GCS Approximation

by

BENJAMIN CROSS

A thesis submitted to
The University of Birmingham
for the degree of
DOCTOR OF PHILOSOPHY

Geometric Modelling Group
School of Mechanical Engineering
College of Engineering and Physical Sciences
The University of Birmingham
Edgbaston
Birmingham B15 2TT
UK

January 2014

Synopsis

This thesis is concerned with the construction of high quality planar curves. The most common representations for curves within Computer Aided Design (CAD) are rational and non-rational polynomials. An inherent problem with polynomials is that a high quality construction can be difficult to achieve. The Generalised Cornu Spirals (GCSs), having a monotonic rational linear curvature profile, are considered high quality curves. However, methods to implement them in existing CAD systems are not currently available. This is partly due to them not being in a polynomial form.

Implementation of the GCS in current CAD systems can be achieved by replacing the GCS with a suitable polynomial approximation. A quintic polynomial approximation is chosen to interpolate the G^2 data of the GCS. This leaves four remaining degrees of freedom to be determined. Values for these four free varaibles will determine the shape and hence quality of the approximation. It will be considered acceptable if the relative error between the curvature profiles of the GCS and approximation is less than a given tolerance.

The main contribution of this thesis is the development of the G^{2+} method, an efficient and robust polynomial approximation to the GCS. Efficiency is achieved by eliminating the requirement of a computationally expensive search. By utilising a subdivision scheme a verification of the robustness will also be achieved. This approximation procedure is therefore worthy of consideration for future CAD implementation.

Acknowledgements

I would like to express my sincerest gratitude to my supervisor Dr R.J. Cripps. I wish to thank him for inspiring me to pursue research and then gifting me the opportunity to do so. His friendly approachability and patience facilitated many thought-provoking conversations which proved to be an invaluable learning resource.

I would also like to express my thanks to all the members of the GMG for their help and support over the years. A special thanks to is owed to Professor A.A. Ball for our brief but stimulating discussions which have led to new thinking and a fresh perspective.

Finally, I wish to give thanks to the School of Mechanical Engineering and to Delcam PLC for providing their much appreciated financial support.

Contents

Chapter 1

Introduction

The discipline of Computer Aided Geometric Design (CAGD) deals with the computational aspects of geometric objects. This thesis is concerned with the construction of one of the most primitive geometric objects, curves. More specifically, it relates to the construction of a high quality planar curve. A planar curve is understood to be a two-dimensional curve, i.e. one that lies on a plane. The Generalised Cornu Spiral (GCS) is a high quality planar curve that is beginning to show value in Computer Aided Design (CAD) and Computer Aided Manufacture (CAM) applications [1–5]. However in its current form it is incompatible with current CAD/CAM systems. This thesis addresses the issue with the development of a robust and efficient polynomial replacement for the GCS.

The chapter begins by discussing what characterises a curve as high quality. It is followed with an overview on the development of high quality curve generation. This can be split into two philosophies: take a curve and apply a *fairing algorithm* to improve its quality, or only

use high quality curves. The GCS is introduced as a suitable choice for the latter of these two approaches. Its incompatibility with CAD/CAM systems motivates the development of a polynomial approximation. This leads to the outline of the research objectives for the thesis. A discussion of existing relevant approximation techniques is then given in a literature review. Finally, this chapter concludes with an outline of the following chapters of the thesis.

1.1 Research motivation

There exists a rich variety of mathematical representations for planar curves, with their practicality dependent upon their application. Within CAD/CAM the most common types are the Non-Uniform Rational B-Splines (NURBS) and the Bézier representations. Since their introduction in the 1970s much research has been dedicated to solving real world problems using these types of curves. The effect of which fortified their place in the CAD/CAM community.

Within the design process of CAD/CAM, a designer will often wish to incorporate geometrical features into their model such as point and tangent interpolation. The designer then requires a sequence of curves, or curve segments, that adhere to this geometrical data. Interpolating different types of geometrical features ensures that the curve segments join with a degree of smoothness. For example, when point data is interpolated there are no gaps inbetween curve segments. Tangent interpolation ensures that segments join even smoother, eliminating tangent discontinuities which appear as an unaesthetic *kink* at the join. After satisfying the geometrical constraints, the shape of the intermediary curve segments is important.

Shape manipulation of these intermediary curves may be possible via adjustments to certain shape parameters, as is the case for β-splines and ν-splines [6]. This procedure is often left to the discretion of the designer. They may refine the curves by hand and decide when a high quality shape has been achieved. However, shape defects can be difficult to diagnose with the naked eye. This issue motivated shape interrogation methods which often involve an analysis of the curvature.

The curvature is an intrinsic geometric property of the underlying curve. Intuitively, one may think of the curvature as a measure of how sharply a curve turns at each point. A more rigorous definition of curvature can be found in section 2.6. Curvature is often used to indicate important geometric features, such as inflection points. These occur when the curvature value is zero and corresponds to locally flat sections of the curve. Similarly, sharp corners can be identified by large curvature regions.

By manipulating the curvature a higher quality curve can be produced. This notion of a high quality curve, with respect to (w.r.t.) shape characteristics, is often described by fairness. Developing a mathematical definition of fairness is problematic, since the measure of quality is often dependent upon the application.

Roulier and Rando observed that "*The notion of fairness is not simply subjective, ambiguous and undefined - it is inherently undefinable*" [7]. Thus, for a curve to be considered fair a more qualitative, rather than quantitative, measure is often used. Such a definition can be found in [8]: "*a curve is fair if its curvature plot is continuous and consists of only a few monotonic pieces*".

The curve representations of Bézier and NURBS, being of a polynomial or rational form, ex-

perience undulating oscillations in curvature [8]. A consequence of this feature is that many curve constructions are deemed unfair. This observation motivated the research of fair curve generation.

Initial approaches involved the use of fairing algorithms. These are processes designed to smooth out the curvature profile by subtly altering the construction of the curves, to reduce some metric related to fairness, yielding a higher quality curve [9]. These processes have an associated computational expense which may have significant downstream effects [8].

An alternative approach to fair curve generation is to restrict the fitting curves used to only high quality curves. One way to ensure high quality is to enforce curvature monotonicity. Examples of this approach include the Pythagorean hodograph spiral [10], the cubic Bézier spiral [11] and the inversion of a hyperbola [12]. However, due to the geometric restriction of monotonic curvature placed upon these curves, there is not always enough flexibility to interpolate sufficient geometrical data between two points with a single curve.

To avoid the undesirable curvature distrubutions from polynomials, curves were constructed entirely from a definition of their curvature profile. Research has been dedicated to the use of Cornu spirals, having linear curvature profiles, to fit to geometric data [13]. However, these curves were also not generally flexible enough to incorporate sufficient geometric data. That is they were only able to interpolate end point and unit tangent data. This motivated the development of the Generalised Cornu Spiral (GCS), an extension to the family of Cornu Spirals. Their curvature profile was defined as a monotonic rational linear function. Their applications to curve segment generation is detailed in [14].

In order to use these curves within CAD/CAM, evaluation of the points is required, a process called *curve synthesis*. A fundamental issue of synthesising a curve directly from a curvature profile derives from the construction process. Due to the intrinsic representation, point evaluation involves an integration step that often yields non-elementary functions. For example, the Cornu spiral requires the evaluation of $\int \sin x^2 \ dx$, referred to as a Fresnel integral [13]. For practical applications, these values are often found via a numerical integration.

Expanding on this issue, the GCS cannot be put into an exact polynomial representation. To argue this, consider any GCS of arbitrary length with non-constant curvature. Take three points on the GCS and interpolate with a circle. As the distance between consecutive points decreases, the interpolating circle converges to the osculating circle at the converged point (see Figure 1.1). For a curve with monotonic curvature the radius of the osculating circles, at varying points along the curve, varies monotonically. Kneser's Theorem [15] states these circles are contained completely within each other (away from inflection points). The limit of these circles, as the radius decreases, has a special property. Any line that intersects this limit circle will intersect the GCS an arbitrary amount of times. Now, from the unisolvence theorem [16], given (n+1) points one can uniquely construct an nth degree interpolating polynomial. Since the line can intersect the GCS an arbitrary amount of times, no finite polynomial can interpolate the GCS at every point. Hence the GCS cannot be represented by a finite polynomial.

As a result, the GCS is incompatible with the polynomial form found in most CAD systems. A compromise can be achieved by replacing the GCS with a non-exact polynomial representation. If the replacement is a sufficiently accurate approximation then it too would demonstrate the fairness properties inhereted from the GCS. This would give rise to a procedure that would help

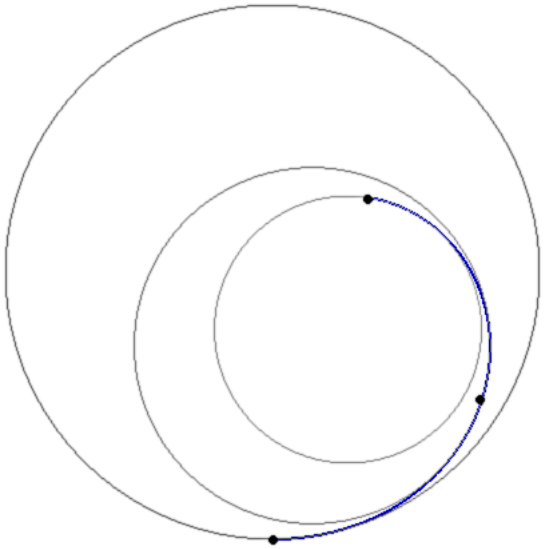

Figure 1.1: Osculating circles at 3 different points of a spiral.

incorporate the GCS into existing systems. This thesis deals with exactly this procedure.

1.2 Research objectives

The goal of this research is to develop a procedure to replace the GCS with a suitable polynomial approximation. To justify its practicality for CAD implementation the procedure should also demonstrate efficiency and robustness.

The first step of the research is to determine what would qualify as a suitable approximation. To begin, the type of polynomial is argued to be the quintic Bézier (section 2.9). There should also be constraints upon the polynomial to ensure essential characteristics are adhered to. These constraints will be argued as the G^2 continuity conditions (section 2.9). Since it is known that the GCS cannot be put into an exact polynomial representation there will be an error associated

with the approximation. To ensure it is a suitable replacement this error should always be within a reasonable tolerance (Chapter 4).

After the procedure has been developed the next step will be to demonstrate efficiency and robustness. Efficiency will be measured as the computational expense required to form an approximation. Robustness will be demonstrated by verifying an acceptable approximation exists for every GCS. The final procedure will be argued to possess both of these properties (Chapter 7).

1.3 Literature review

This section presents a review of existing research that is relevant to the objectives discussed in section 1.2. Since this thesis is concerned with the approximation of a GCS, the literature review discusses some existing curve approximation techniques. Very little research has been specifically dedicated towards polynomial approximations of the GCS. Thus degenerate curve types of the GCS are also examined. These include the circle, the logarithmic spiral and the Cornu spiral (section 3.1). The different approaches taken are analysed with respect to their compatibility to GCS approximation. Issues to consider include approximation accuracy, geometric restrictions, computational efficiency and robustness of the algorithm.

An approximation's accuracy will be considered acceptable if the relative error between the curvature profiles of the GCS and approximation does not exceed 5%. The formal measurement of this error as well as the reasoning behind the value of 5% can be found in Chapter

4. Geometric restrictions considered involve determining the ability to extend each method to approximate a general GCS. Approximations to the GCS will also have to satisfy certain geometrical constraints, the G^2 conditions. This involves matching end position, tangent direction and curvature values (see section 2.8.2 for more details). Ensuring the G^2 constraints guarantees a degree of smoothness, or level of continuity, between the joints of the curve segments. The first two conditions (position and tangent direction interpolation) ensure there are no gaps or kinks in the curve (see section 1.1). Matching curvature values ensures that the curvature profile of the whole curve is continuous i.e. no gaps (see section 2.9).

The efficiency of each method will be measured by the computational expense required to form an approximation. Seeking a quantifiable measure for efficiency could be done by analysing the computational time required to construct the approximation (or alternatively via the number of floating point operations (FLOPS) required). However, more of a qualitative analysis will be given as this is enough to argue practicality of the method with respect to the goals of this thesis.

Finally, a method is considered robust if the underlying construction algorithm always yields acceptable approximations. If an approximation is not acceptable then the method should have a built in procedure to improve the accuracy, eventually leading to the formation an acceptable approximation.

1.3.1 Circle approximations

Goldapp [17] gives an approximation to an arc of the unit circle, with a spanning angle $\theta \leq \frac{\pi}{2}$, using a cubic Bézier curve (see Fig 1.2). Note that any segment of any circle can then be approximated from this method by appropriate joining and scaling of the Bézier approximates.

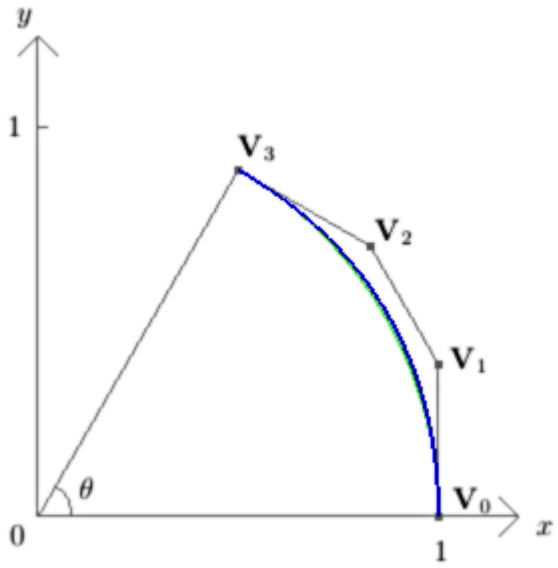

Figure 1.2: Approximating a circular arc with a cubic Bézier

To begin the approximations are constructed to adopt the symmetry possessed by the circular segment. Three different approximation methods are given: a G^1 interpolation, a C^0 interpolation and a non-C^0 interpolation of the end points. The G^1 interpolation ensures position and unit tangent continuity with the circle at the end points, leaving one degree of freedom. The C^0 interpolation gains another degree of freedom by allowing the start and end tangents to point in any direction, but retain symmetry. Finally, another degree of freedom is introduced by letting

the start and end points lie on the line from end point to the centre of the circle. This is the non-C^0 method.

Using the free parameters an error function is then minimised by studying stationary points and making the extreme errors equi-oscillate. The error function is the distance from the circle,

$$\rho(t) = \sqrt{x^2(t) + y^2(t)} - 1$$

which is simplified to

$$\epsilon(t) = x^2(t) + y^2(t) - 1.$$

with

$$\epsilon_{max} = \max_{t \in [0,1]} |\epsilon(t)|.$$

Analysis when approximating a quarter circle gives errors of $\epsilon_{max} \approx 10^{-4}$ for the C^0 interpolations and a slight improvement but still of the same order for non-C^0 interpolations. The G^1 is the least accurate with an error approximately 5 times larger.

- **Approximation accuracy** Analysis of the curvature is presented corresponding to a relative curvature difference of around 0.005. Although this is acceptable in the context of a GCS, approximations to the circle can be much more accurate. The order of this polynomial approximation is cubic. Therefore a higher degree of Bézier could be used to

improve accuracy.

- **Geometric restrictions** Extending this technique to the GCS is not possible. This is because although the process of minimising the errors of $\epsilon(t)$ is appropriate for a circle, no such simple form exists for the general GCS. Also, it will be argued that a GCS approximation should be a G^2 interpolation (section 2.7) of which this method is not.

- **Computational efficiency** The most computationally expensive process involves minimising the error function. This is shown to be equivalent to a system of three nonlinear equations which is then solved numerically. This process is relatively inexpensive.

- **Robustness** An arbitrary spanning angle of the circle, $\theta \leq \frac{\pi}{2}$, means that if an improvement of accuracy is required then the circular section can be split. This would result in an improved approximation in the smaller curve approximations to the subdivided circle segment.

Fang approximated a circular arc using a quintic Bézier [18]. Assuming a symmetric model with at least a G^1 interpolation with the circle leaves three free parameters (p, q, r), see Fig 1.3.

Using these three free parameters, Fang defined seven different approximation methods. The seven approximations are similar in construction in that they all interpolate geometric data at the start, middle and end points. Analysis for three of these approximations will be discussed. The four other approximations are similar in construction to these three cases and have been omitted for brevity. The concluding remarks are equally valid for all seven approximations. Adopting Fang's nomenclature the three methods of interest are

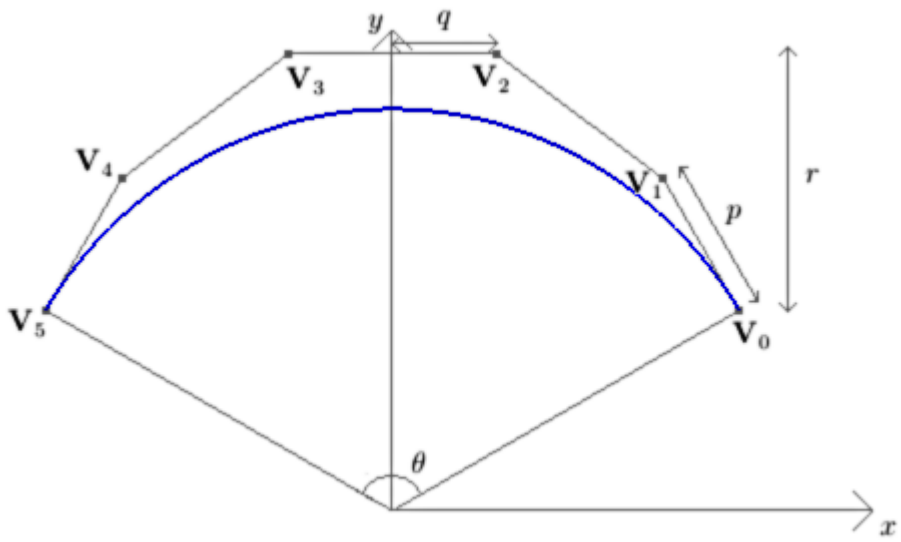

Figure 1.3: Approximating a circular arc with a quintic Bézier

- A G^2 interpolant of the beginning, middle and end point - ($\mathbf{II} - p_5$).
- A G^3 interpolant of the start and end point together with mid-point position (G^0) interpolation - ($\mathbf{III} - p_8$)
- A G^4 interpolation of the start and end points - ($\mathbf{V}$).

These different methods were then analysed with respect to three error functions (where $\kappa(t)$ denotes the curvature at t and s represents arc-length).

$$\epsilon_r(t) = \sqrt{x^2(t) + y^2(t)} - 1 = \rho(t)$$

$$\epsilon_\kappa(t) = |\kappa(t)| - 1$$

$$\epsilon_v(t) = \frac{d}{ds}|\kappa(t)|$$

$Method$	$\lvert\epsilon_r\rvert_{max}$	$\lvert\epsilon_\kappa\rvert_{max}$	$\lvert\epsilon_v\rvert_{max}$
$\mathbf{II} - p_5$	1.22×10^{-8}	7.74×10^{-7}	3.33×10^{-5}
$\mathbf{III} - p_8$	2.95×10^{-8}	1.18×10^{-6}	1.07×10^{-5}
$\mathbf{V}$	3.68×10^{-7}	5.68×10^{-6}	3.29×10^{-5}

Table 1.1: Table of errors for quarter circle approximations.

Values for these error functions as an approximation to the quarter circle are given in Table 1.1.

- **Approximation accuracy** The relative curvature difference error is around 10^{-6} and is thus acceptable.

- **Geometric restrictions** This method presents an effective technique to G^2 and G^3 end point interpolation which will be useful for polynomial constructions to the GCS. This method exploits a geometrical property of the circle, symmetry. Matching derivative data at the midpoint is relatively simple via rules of symmetry. However, this does not extend to any non-circular GCS as these curves are non-symmetrical. Mid-point derivative data interpolation therefore becomes a much more difficult task.

- **Computational efficiency** The methods outlined are constructed using algebraic expressions and no numerically expensive processes are required.

- **Robustness** Similarly to [17] an arbitrary spanning angle means that improvements to accuracy can be found by splitting the circular segment.

1.3.2 Logarithmic spiral approximations

Baumgarten and Farin developed a piece-wise approximation to a logarithmic spiral segment in [19]. The method uses rational cubic Bézier curves. First the G^2 constraints at the end points are satisfied then, with the free parameters, the equiangular property of the spiral is approximated. The equiangular property states that the angle between the tangent at a point and the ray from the point through the origin is equal for all points on the spiral (see Figure 1.4).

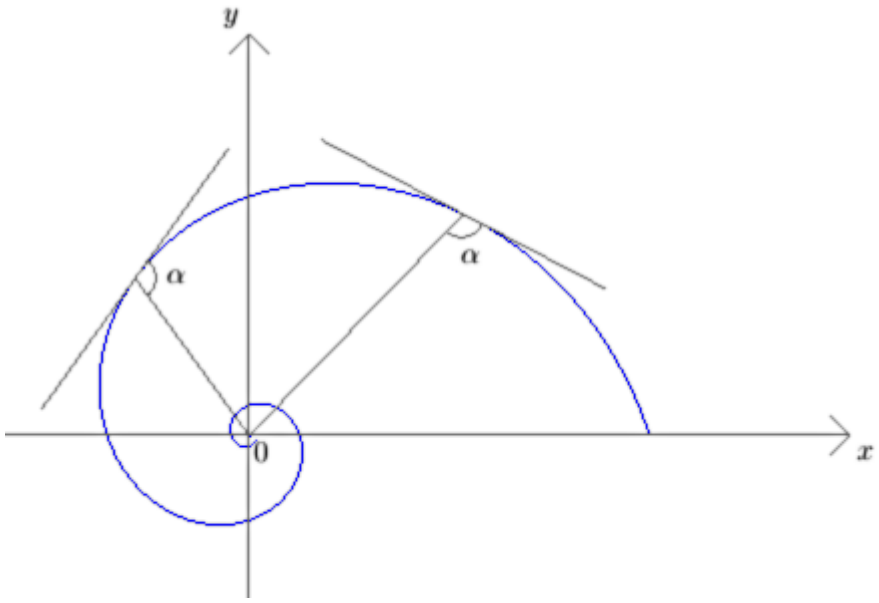

Figure 1.4: Equiangular property of a logarithmic spiral

The total degrees of freedom for a rational cubic are twelve, four each for the w_i and the x and y values for the $\mathbf{V}_i$. Two weights may be chosen arbitrarily as outlined in [8] and are set as $w_1 = w_2 = 2/3$ to help algebraic manipulations. Eight degrees of freedom are required to interpolate the G^2 constraints at the end points. The remaining two degrees of freedom are left in the form of the tangent vector lengths $|\mathbf{V}_1 - \mathbf{V}_0|$ and $|\mathbf{V}_2 - \mathbf{V}_3|$.

The equiangular property is expressed with the conditions:

$$\frac{|(\mathbf{V}(t)-\mathbf{o})\times\mathbf{V}'(t)|}{(\mathbf{V}(t)-\mathbf{o})\cdot\mathbf{V}'(t)}=\frac{\sin(\alpha)}{\cos(\alpha)}=\tan(\alpha),$$

where $\mathbf{o}$ is the origin of the logarithmic spiral. The two free parameters are then chosen to best approximate this property.

Analysis of the approximation method is given. It is shown that the error function (a comparison of arc lengths) will satisfy a given tolerance after a suitable number of subdivisions.

- **Approximation accuracy** Analysis of the curvature indicates a relatively poor match (an error much greater than 5%) for the example given in [19].
- **Geometric restrictions** This method is novel in the fact that it uses geometrical arguments to construct its approximation. The equiangular property however does not extend to a general GCS. Approximations do satisfy the G^2 constraints by construction.
- **Computational efficiency** The construction process is relatively inexpensive. However, in order to achieve a satisfactory approximation many relatively small segments may be used leading to data proliferation [2].
- **Robustness** A strength of the method is the ability to easily subdivide the logarithmic spiral. Applying smaller approximations is a feasible approach for the relatively simple logarithmic spiral since it can be expressed by:

$$\mathbf{L}(\lambda)=\begin{bmatrix}x(\lambda)\\ y(\lambda)\end{bmatrix}=\begin{bmatrix}r_0e^{k\lambda}\cos(\lambda)\\ r_0e^{k\lambda}\sin(\lambda)\end{bmatrix}.$$

No such form, in terms of elementary functions, exists for the general GCS. Furthermore, the process of GCS subdivision has not been formalised. This notion of GCS subdivision will be addressed inside this thesis and can be found in section 3.5.

1.3.3 Cornu spiral approximations

Perhaps the earliest approximation to the Cornu spiral that was used extensively in CAD was the rational approximations provided by Heald [20]. The approach recasts the formula for a Cornu spiral in polar form and minimises the polar radial difference between approximation and spiral.

The approximation can then be converted back into parametric form to yield approximations to the Fresnel integrals. The resulting approximation is:

$$\begin{bmatrix} x(t) \\ y(t) \end{bmatrix} = \begin{bmatrix} \frac{1}{2} - R(t)\sin(\frac{1}{2}\pi(A(t) - t^2)) \\ \frac{1}{2} - R(t)\cos(\frac{1}{2}\pi(A(t) - t^2)) \end{bmatrix},$$

where

$$\begin{aligned} R(t) &= \frac{0.506t + 1}{1.79t^2 + 2.054t + \sqrt{2}}, \\ A(t) &= \frac{1}{0.803t^3 + 1.886t^2 + 2.524t + 2}. \end{aligned} \tag{1.1}$$

- **Approximation accuracy** Analysis of the curvature is not considered.

- **Geometric restrictions** End point interpolation is not guaranteed with this approximation. Since the GCS approximation will require G^2 interpolation, this approach will not be acceptable.

- **Computational efficiency** The evaluation of points is relatively fast and thus can be considered efficient.

- **Robustness** There is no way to improve the accuracy and so it cannot be considered robust.

A more robust method to Cornu spiral approximation was developed by Wang et. al [21] considering Taylor expansions of the Fresnel integrals. The idea was to take a Taylor series of the Cornu spiral about the start and end points independently. Then a Hermite style blend of the two functions could produce a G^2 interpolating function, providing the order of the Taylor series was taken to a sufficient degree.

The result is a complex infinite series which can be truncated and put into B-spline form via the Hermite blending functions. A method to produce a single Bézier representation is given but this function is not even G^0 (does not interpolate end points).

- **Approximation accuracy** Since the accuracy can always be improved by taking a higher order Taylor series, the level of accuracy is not an issue.

- **Geometric restrictions** The constructions are more algebraically driven than geometrical and an extension to a general GCS in this format is achievable. A G^2 construction is possible.

- **Computational efficiency** This method is not practical since at least a degree 8 Bézier is required for a G^0 interpolation. The calculations required to produce the G^2 interpolant are very intensive and hence impractical.

- **Robustness** The accuracy can always be improved by taking a higher order Taylor series.

Sánchez-Reyes constructed polynomial approximations via s-power series [22]. These approximations essentially interpolated the parametric derivative data at the end points of the curve up to an arbitrary degree. The quintic polynomial case corresponds to a C^2 interpolation at end points. This curve may also be referred to as the second order Hermite interpolant. It will be used later on in the thesis as a means of comparison with existing methods (see Chapter 8).

- **Approximation accuracy** Arbitrary accuracy can be achieved by increasing the order of the polynomial.

- **Geometric restrictions** The constructions are more algebraically driven than geometrical and extensions to a general GCS in this format is achievable. A G^2 construction exists providing the degree of the polynomial is at least five.

- **Computational efficiency** To achieve an acceptable approximation the order may be too high to be considered practical for CAD/CAM use.

- **Robustness** Approximation accuracy can be improved by increasing the degree of the approximating polynomial.

1.3.4 Generalised Cornu Spiral (GCS) approximations

Relatively little work has been carried out into polynomial approximations of a general GCS. A method using a quintic Bézier as an interpolating function is described by Cripps et. al [23] which follows on from the work by Zhu [24].

First a tolerance is defined by considering errors corresponding to approximations of two bounding circles. Then the G^2 conditions are satisfied leaving four free parameters. A search algorithm is applied to these variables until an error less than the tolerance is achieved. This error was defined as the relative difference in curvature when $|\kappa(s) > \mu|$ for some tolerance μ and the absolute difference in curvature otherwise. This approach was novel in the sense that most approximations would measure the error w.r.t. to the position of the points on the curve whereas this method considered the curvature of points on the curve. It was established that acceptable polynomial approximations do exist.

- **Approximation accuracy** In the analysis of the method, results show the approximation to reflect the curvature profile of the GCS accurately.
- **Geometric restrictions** G^2 conditions are satisfied.
- **Computational efficiency** The original work in [24] uses FEP (Fast Evolutionary Programming) to converge to a solution. Since no geometrical features of the construction are taken advantage of, the method may be interpreted as a non-linear four dimensional optimization problem. The computational cost required before finding a satisfactory solution is not discussed. However, the presence of a search algorithm itself is indicative of relatively poor efficiency.

- **Robustness** The GCS approximations given in [23] and [24] did not consider approximations of arbitrary precision. Furthermore, it is impossible to guarantee an acceptable approximation will be found when using a search algorithm.

1.3.5 Summary

This thesis uses similar techniques discussed in [18] to develop G^2 and G^3 constructions using quintic Bézier curves. Robustness in [19] was achieved via a process of subdividing the curve. This idea will be extended to a general GCS in section 3.5. The thesis will also build upon the ideas developed in [23] and [24], eventually removing the necessity of a numerical search. This in turn will help demonstrate efficiency and robustness. The type of error used here will inspire a new metric, measuring discrepancies in curvature as opposed to using position based-metrics. This will be developed in Chapter 4. The s-power series of [22] will be useful as a means of comparison to any developed approximation methods since general GCS approximations are possible.

1.4 Outline of thesis

The following chapter introduces the relevant background theory necessary for the subsequent chapters. This includes formal definitions of the Bézier curve, curvature and geometric continuity. The third chapter examines the GCS in more detail. Important properties are established that will be used further into the thesis. Chapter 4 introduces the error metric used to mea-

sure the accuracy of an approximation. In Chapter 5 the first approximation technique, the G^3 method, is described. The chapter ends with a discussion on the drawbacks which leads to the development of the G^{2+} method, detailed in chapter 6. Chapter 7 verifies the robustness of the G^{2+} method with a numerical analysis procedure explained within. Chapter 8 then illustrates some examples of approximations using the developed G^3 and G^{2+} models. The final chapter concludes with a summary on the research contributions of the thesis, finishing with some remarks on possible future research.

Chapter 2

Background Information

In this chapter the relevant background theory is presented. It begins by looking at the definition of planar curves. This is followed with an extensive look at the Bézier form. Then reparameterisations of a curve are discussed which leads to the important concept of *arc-length parameterisation*. Taking this further, the formal definition of curvature and curvature synthesis is presented.

It is then possible to study the GCS from a more rigorous viewpoint. Following this, the theory behind interpolation of the GCS is studied by first looking at parametric continuity and then geometric continuity. In the final section, the aims of the thesis are recast inside a more mathematically rigorous framework.

2.1 Planar curves

A planar curve is a curve that lies in a plane. That is to say it is a two-dimensional curve. It is most often described as a collection of points in the $x - y$ plane. An efficient way to construct a planar curve on a computer is to use the parametric form [8]. The values for the (x, y) coordinates are given by a function

$$\mathbf{V}(t) = \begin{bmatrix} x(t) \\ \\ y(t) \end{bmatrix} \qquad t \in [a, b].$$

The vector $\mathbf{V}(a)$ is referred to as the start point and $\mathbf{V}(b)$ the end point. A stable and efficient function type for $\{x(t), y(t)\}$ are polynomials. In this form the curve may be represented by

$$\mathbf{V}(t) = \begin{bmatrix} x(t) \\ \\ y(t) \end{bmatrix} = \begin{bmatrix} \sum_{i=0}^{n} x_i t^i \\ \\ \sum_{i=0}^{n} y_i t^i \end{bmatrix} = \sum_{i=0}^{n} t^i \begin{bmatrix} x_i \\ \\ y_i \end{bmatrix} = \sum_{i=0}^{n} t^i \mathbf{V}_i \quad t \in [a, b]. \tag{2.1}$$

The curve is said to be of degree n or order $n + 1$. The $n + 1$ vectors, $\mathbf{V}_i$, are referred to as the control points and define the shape of the curve [8].

2.2 Transformations of a curve

Given a curve it is possible to find transformations which preserve the shape but alter its position and orientation. These consist of translations and rotations. These types of transformations are

known as *rigid body transformations* or *isometries* [25]. Combining isometries with a scaling factor gives the *proportional scaling transformations*. These are all examples of *affine transformations* with the exception that affine transformations also include *non-proportional scaling*. The components of these transformations are given in the following equations:

Translation:	$T_{a,b}$	$(x, y) \mapsto (x + a, y + b),$
Rotation:	R_θ	$(x, y) \mapsto (x\cos\theta - y\sin\theta, x\sin\theta + y\cos\theta),$
Proportional scaling:	S_a	$(x, y) \mapsto (ax, ay),$
Non-proportional scaling:	$S_{a,b}$	$(x, y) \mapsto (ax, by).$

2.3 The Bézier curve

When using polynomial curves it can often be convenient to express them in a different form other than (2.1). For example, one might want to define the curve over the interval $[0, 1]$. This can be achieved by a transformation of the parameter $u = (t-a)/(b-a)$. One may also decide to use a different basis for the polynomial which can provide more insight into the curve.

One such basis is called the Bernstein basis [8] and is given as

$$\begin{aligned} B_i^n(t) &= \binom{n}{i}(1-t)^{n-i}t^i \\ &= \frac{n!}{i!(n-i)!}(1-t)^{n-i}t^i. \end{aligned}$$

for $i = 0, 1, \ldots, n$. By choosing the basis functions as the Bernstein polynomials, geometric properties of the curve may be inferred immediately. This results in the definition of the Bézier

curve.

2.3.1 Definition

The Bézier curve of degree n is defined on $[0, 1]$ as the polynomial with the Bernstein basis functions and is given by

$$\mathbf{V}(t) = \sum_{i=0}^{n} B_i^n(t)\mathbf{V}_i \quad t \in [0, 1]. \tag{2.2}$$

The Bézier form is useful in curve construction as it has many useful geometrical properties.

2.3.2 Properties

- *Endpoint interpolation*. The start of the curve is given by the initial control point $\mathbf{V}_0$ and the end point is given by $\mathbf{V}_n$. This can be verified easily by substitution into (2.2).

- *Derivative control*. It can be shown that the k-th derivative of the Bézier is given by [8]

$$\frac{d^k\mathbf{V}(t)}{dt^k} = \frac{n!}{(n-k)!}\sum_{i=0}^{n-k}\left(\sum_{j=0}^{k}\binom{k}{j}(-1)^{k-j}\mathbf{V}_{i+j}\right)B_i^{n-k}(t).$$

  As a consequence, the k-th derivative at the start of the curve is completely defined by the first $k + 1$ control points. Similarly, the k-th derivative at the end point is completely defined by the last k control points. So for example: the first derivative of the curve at the start point, which corresponds to the initial tangent, is given by $\mathbf{V}'(0) = n(\mathbf{V}_1 - \mathbf{V}_0)$.

(The notation $f'(x)$ refers to the derivative of the function w.r.t. its parameter, i.e. $\mathbf{V}'(t) = \frac{d}{dt}\mathbf{V}(t)$. Similarly, $f''(x)$ and $f'''(x)$ will be used interchangeably as $f''(x) = \frac{d^2}{dx^2}f(x)$ and $f'''(x) = \frac{d^3}{dx^3}f(x)$ throughout the thesis.)

- *Affine invariance*. This property means that if an affine transformation is applied to a curve, the resulting curve will also be a Bézier. This can be verified easily by applying the affine transformation to the control points [8].

An example of a cubic Bézier curve is given in Figure 2.1.

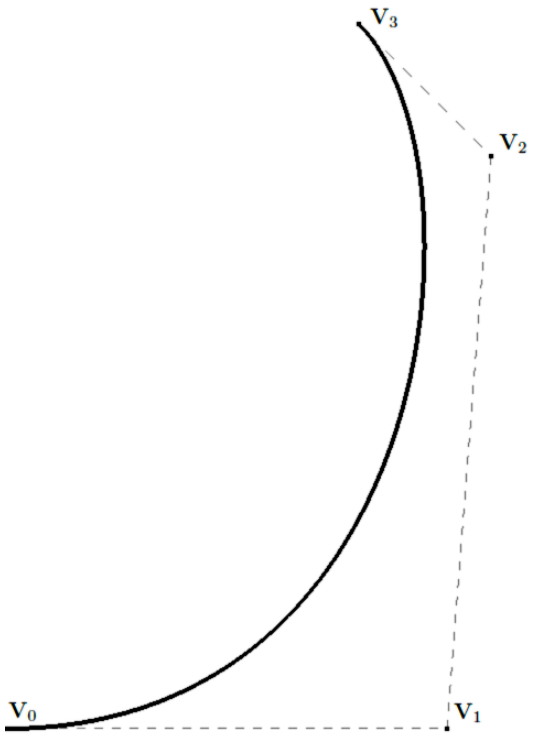

Figure 2.1: A cubic Bézier curve.

2.4 Rational vs. non-rational representations

The most common types of curve representation used within the CADCAM environment are polynomial based. These can be split into two categories: polynomial and rational functions; a rational function is the quotient of two polynomials. Ordinary polynomials (or integral polynomials) are represented with polynomial basis functions $B_i^n(t)$; each function is a multiplier of a control point $\mathbf{V}_i$ as in (2.3). Rational polynomials have a set of basis functions that are rational polynomial which are multiplied by the control point $\mathbf{V}_i$ as in (2.3).

$$\begin{aligned} \text{Non-rational form:} \quad & \mathbf{V}(t) = \textstyle\sum_{i=0}^{n} B_i^n(t)\mathbf{V}_i \\ \text{Rational form:} \quad & \mathbf{V}(t) = \textstyle\sum_{i=0}^{n} \dfrac{w_i B_i^n(t)}{\sum_{i=0}^{n} w_i B_i^n(t)}\mathbf{V}_i \end{aligned} \tag{2.3}$$

Rational curves have a distinct advantage over their non-rational counterparts. Using the rational form serves to increase the degrees of freedom by assigning values for the weight parameters w_i. For quadratic polynomials these values hold a geometric significance. The rational quadratic curves can be observed as a projection of the curve that is defined by values of w_i. A consequence being that rational quadratic curves are able to represent all conic sections exactly. This property helped establish their usage within the CAD/CAM community.

However, as the degree of the polynomial surpasses two, the w_i's are no longer comprehensible by this geometric interpretation. Instead the weights are often determined heuristically. Rational representations are often avoided to eliminate floating point errors. These arise as a consequence from the division operation that differentiates them from their non-rational counterpart. By increasing the complexity, certain algebraic manipulations are no longer possible with a rational

curve representation yet are permitted with the non-rational form. The GCS is not a conic section (with exception to the circle) and thus choosing rational polynomials to approximate it will result in an increase in complexity (w.r.t. algebraic manipulation). The non-rational polynomial form is therefore a more suitable initial choice for consideration. Rational curves will not be considered in this thesis but may be of interest for future research (see section 9.4).

2.5 Curve re-parameterisation

A parametric curve is described with a vector function, $\mathbf{V}(t)$, and a parameter range $t \in [a, b]$. The mathematical description of the parameter t may be altered to $u(t)$ yielding a re-parameterisation of the curve as $\mathbf{V}(u(t))$. An example of a re-parameterisation, which changed the domain of a parametric equation from $t \in [a, b]$ to $u \in [0, 1]$ and was the transformation $u(t) = (t - a)/(b - a)$, as observed in section 1.2. Applying this transformation did not affect the shape of the curve and is consequently referred to as an equivalent parameterisation.

2.5.1 Equivalent parameterisations

Any re-parameterisation that describes the same oriented curve is said to be an *equivalent parameterisation*. If a re-parameterisation, u, is given by

$$u : [a, b] \to [c, d]$$

$$t \mapsto u(t)$$

then it was shown in [28] to be equivalent if

$$\begin{aligned} &(i) \quad \mathbf{V}(t) = \mathbf{V}(u(t)) \quad \forall t \in [a,b] \\ &(ii) \quad u([a,b]) = [c,d] \\ &(iii) \quad \frac{d}{dt}u(t) > 0 \quad \forall t \in [a,b]. \end{aligned} \tag{2.4}$$

An important intrinsic property of curves that is not affected by equivalent parameterisations is the length of the curve.

2.5.2 Arc-length parameterisation

Given a parametric curve $\mathbf{V}(t)$, $t \in [a,b]$ it is possible to find the length of the curve, S, from

$$S = \int_a^b |\mathbf{V}'(\tau)| d\tau = \int_a^b \sqrt{(x'(\tau)^2 + y'(\tau)^2)} d\tau.$$

This leads to the unique re-parameterisation called the *arc-length parameterisation* where the new parameter, s, gives the length of the curve. This re-parameterisation can be found by calculating the arc-length at a point on the curve from

$$s(t) = \int_a^t |\mathbf{V}'(\tau)| d\tau = \int_a^t \sqrt{(x'(\tau)^2 + y'(\tau)^2)} d\tau. \tag{2.5}$$

Since $s(t)$ is the cumulative integral of a positive function, $|\mathbf{V}'(\tau)|$, it is a monotonic increasing function on $[0, S]$. Thus there exists an inverse function $s^{-1}(t)$ [26]. Using this to re-

parameterise the curve the arc-length parameterisation is given by $\mathbf{V}(s^{-1}(t))$.

Whenever a curve is arc-length parameterised it is usually expressed as $\mathbf{V}(s)$ where s is the arc-length parameter with $s \in [0, S]$. Using this form for curve representation intrinsic properties of the curve can be studied in further detail. One such property, curvature, is discussed next.

2.6 Curvature

Consider an arc-length parameterised curve, $\mathbf{V}(s)$. Since the parameter s must reflect arc length, using

$$\int_0^t |\mathbf{V}'(\tau)| d\tau = s(t) = s$$

it must be that $|\mathbf{V}'(s)| \equiv 1$ and thus $\mathbf{V}'(s)$ represents the unit tangent vector $\mathbf{T}(s)$. The unit tangent vector may be expressed as [26]

$$\mathbf{V}'(s) = \mathbf{T}(s) = \begin{bmatrix} \cos(\theta(s)) \\ \sin(\theta(s)) \end{bmatrix} \tag{2.6}$$

where $\theta(s)$ is the angle of the tangent vector at s.

Furthermore,

$$\mathbf{T}(s) \cdot \mathbf{T}(s) = |\mathbf{T}(s)|^2 = 1$$

and thus

$$\frac{d}{ds}(\mathbf{T}(s) \cdot \mathbf{T}(s)) = 2\mathbf{T}(s) \cdot \mathbf{T}'(s) = 0$$

therefore the vectors $\mathbf{T}(s)$ and $\mathbf{T}'(s)$ are perpendicular. This means that $\mathbf{T}'(s)$ lies along the unit normal vector, $\mathbf{N}(s)$, and so

$$\mathbf{T}'(s) = \kappa(s)\mathbf{N}(s).$$

The value $\kappa(s)$ is referred to as the curvature at s. By differentiating the unit tangent (2.6) w.r.t. arc-length:

$$\mathbf{T}'(s) = \theta'(s)\mathbf{N}(s)$$

and thus

$$\kappa(s) = \theta'(s). \tag{2.7}$$

Curvature is therefore the rate of change of the tangent angle along the curve w.r.t. arc-length (i.e. travelling at a uniform speed). Intuitively, one can think of curvature as measure of how much a curve *curves* at a point.

A possible approach to obtain the curvature may be to use the fact that $\kappa(s) = |\mathbf{V}''(s) \times \mathbf{V}'(s)|$, where "$\times$" represents the vector cross product. However, finding an analytical expression for the arc-length parameterisation of a curve is not always possible. This is because the inverse of the

arc-length function from (2.5), $s^{-1}(t)$, is not always an analytical function. Furthermore, it was proved in [27] that polynomial curves do not have a polynomial arc-length parameterisation.

The curvature of a curve that is not parameterised by arc-length can be calculated from the following method. To begin, observe

$$\tan(\theta) = \frac{dy}{dx} = \frac{y'(t)}{x'(t)}$$

Then

$$\begin{aligned}
\frac{d}{dt}\tan(\theta) &= \frac{x'(t)y''(t) - y'(t)x''(t)}{x'(t)^2} \\
&= \sec^2(\theta)\frac{d\theta}{dt} = \left[1 + \tan^2\theta\right]\frac{d\theta}{dt} = \left[1 + \left(\frac{y'(t)}{x'(t)}\right)^2\right]\frac{d\theta}{dt} \\
\frac{d\theta}{dt} &= \frac{x'(t)y''(t) - y'(t)x''(t)}{x'(t)^2 + y'(t)^2}
\end{aligned} \tag{2.8}$$

From (2.5) observe that:

$$\begin{aligned}
s(t) &= \int_0^t |\mathbf{V}'(\tau)| d\tau \\
\frac{ds}{dt} &= |\mathbf{V}'(t)| = \left(x'(t)^2 + y'(t)^2\right)^{\frac{1}{2}}.
\end{aligned} \tag{2.9}$$

Thus combining (2.7), (2.8) and (2.9) [26]:

$$\begin{aligned}
\kappa(t) &= \frac{d\theta}{ds} = \frac{d\theta}{dt}\frac{dt}{ds} \\
&= \frac{x'(t)y''(t) - y'(t)x''(t)}{(x'(t)^2 + y'(t)^2)^{\frac{3}{2}}}.
\end{aligned} \tag{2.10}$$

2.6.1 Curvature synthesis

To ensure fairness of a curve, the curvature should be controlled. However the curvature profile of a polynomial is notoriously difficult to manipulate [9]. This can be seen as a consequence of the form of equation (2.10). As an alternative to polynomials, a curve may be *synthesised* directly from a curvature profile.

By using (2.6) and (2.7) it is possible determine the curve $\mathbf{V}(s)$ by twice integrating to give

$$\mathbf{V}(s) = \begin{bmatrix} x(s) \\ y(s) \end{bmatrix} = \begin{bmatrix} x_0 + \int_0^s \cos(\theta_0 + \int_0^\sigma \kappa(\tau)d\tau)d\sigma \\ y_0 + \int_0^s \sin(\theta_0 + \int_0^\sigma \kappa(\tau)d\tau)d\sigma \end{bmatrix} \tag{2.11}$$

accounting for initial conditions x_0, y_0, θ_0. The values of (x_0, y_0) correspond to the start point of the curve, and θ_0 corresponds to the angle the initial tangent forms with the positive x-axis.

Therefore it is possible to define a curve directly from its curvature profile, thus ensuring fairness. This inspired the development of the generalised Cornu spiral.

2.7 The Generalised Cornu Spiral (GCS)

2.7.1 Definition

The generalised Cornu spiral was defined as a curve with a rational linear monotonic curvature profile given by [14]:

$$\kappa(s) = \frac{p + qs}{S + rs} \qquad s \in [0, S] \qquad r > -1.$$

The value of r is restricted to $r > -1$ so the function is well defined.

2.7.2 Monotonicity of curvature

The curvature monotonicity can be shown by considering

$$\kappa'(s) = \frac{Sq - pr}{(S + rs)^2}$$

thus $\kappa'(s) \neq 0$ except when $Sq = pr$. If $Sq = pr$ then $q = \frac{pr}{S}$ and so

$$\kappa(s) = \frac{p + \frac{pr}{S}s}{S + rs} = \frac{\frac{p}{S}(S + rs)}{(S + rs)} = \frac{p}{S}$$

and thus the curvature is constant. It then follows that the generalised Cornu spiral has a monotonic curvature profile and can therefore be considered a fair curve.

2.7.3 GCS synthesis

From the curvature profile it is possible to synthesize the GCS using (2.11) to obtain:

$$\mathbf{F}(t) = \begin{bmatrix} x(t) \\ y(t) \end{bmatrix} = \begin{bmatrix} x_0 + \int_0^t \cos\left[\theta_0 + \int_0^\sigma \frac{p+q\tau}{S+r\tau} d\tau\right] d\sigma \\ y_0 + \int_0^t \sin\left[\theta_0 + \int_0^\sigma \frac{p+q\tau}{S+r\tau} d\tau\right] d\sigma \end{bmatrix}$$

which gives

$$\begin{bmatrix} x(t) \\ y(t) \end{bmatrix} = \begin{cases} \begin{bmatrix} x_0 + \int_0^t \cos\left[\theta_0 + \frac{2p\sigma + q\sigma^2}{2S}\right] d\sigma \\ y_0 + \int_0^t \sin\left[\theta_0 + \frac{2p\sigma + q\sigma^2}{2S}\right] d\sigma \end{bmatrix} & \text{if } r = 0, \\ \begin{bmatrix} x_0 + \int_0^t \cos\left[\theta_0 + \frac{rq\sigma + (pr - qS)\ln(1 + \frac{r\sigma}{S})}{r^2}\right] d\sigma \\ y_0 + \int_0^t \sin\left[\theta_0 + \frac{rq\sigma + (pr - qS)\ln(1 + \frac{r\sigma}{S})}{r^2}\right] d\sigma \end{bmatrix} & \text{otherwise.} \end{cases} \tag{2.12}$$

Analytical solutions to these integrals (2.12) cannot be found in general. Instead they are usually left in the integral form and values obtained by numerical integration [14]. As a consequence this representation is impractical for direct use within CAD.

A compromise between a Bézier curve and the GCS is needed. When constructing Bézier curves it is not possible to control the curvature and hence ensure fairness. The GCS, although high quality, is not implementable in the CAD environment. A suitable solution is to replace the GCS with a Bézier curve. Since a general GCS is not a polynomial [19, 23] the Bézier would have to be an approximation that mimics the GCS sufficiently.

This Bézier approximation should satisfy certain conditions to justify it as a suitable replacement. For example, one condition to be imposed could be that the end points of the Bézier must agree with the GCS. This can be achieved by equating position data at the start and end points of the curves. Further characteristics of the Bézier approximation may be constrained to match that of the GCS. This process is described as interpolation.

2.8 Continuity of interpolation

End point interpolation of curves is important to ensure the connection between two curves is continuous, i.e. there are no gaps. When joining two Bézier curves together this property is essential. Extending this notion to first derivatives, which relate to tangents, a smooth transition across a join means there are no kinks in the curve and thus a level of first derivative continuity must be ensured.

An interpolation will achieve a certain level of continuity dependent upon the degree of derivative that is matched. To ensure a sufficient level of continuity between two curves continuity conditions are imposed.

2.8.1 Parametric continuity

Let $\mathbf{P}(t)$ and $\mathbf{Q}(u)$ be curves. The curves are said to join with C^n derivative continuity at a if

$$\frac{d^i\mathbf{P}}{dt^i}(a) = \frac{d^i\mathbf{Q}}{du^i}(a) \qquad \text{for } i = 0 \ldots n$$

i.e. their derivatives match.

If the derivatives of a curve (up to n-th order) at the start and end points are given, the unique polynomial (of degree $2n+1$) that interpolates these derivatives is called the Hermite interpolant [22].

However, since only the shape of a curve is important, re-parameterisation should not affect the level of continuity. This is not generally the case for C^n continuity [28] and thus a different type of continuity should be considered.

2.8.2 Geometric continuity

Let $\mathbf{P}(t)$ and $\mathbf{Q}(u)$ be two curves. They are said to be G^n derivative continuous at a if there exists an equivalent parameterisation, $v(u)$, for u such that [28]

$$\frac{d^i\mathbf{P}}{dt^i}(a) = \frac{d^i\mathbf{Q}}{dv^i}(v(a)) \qquad \text{for } i = 0 \ldots n.$$

This type of continuity is parameterisation invariant. Thus given a G^n continuous curve, it is possible to re-parameterise without affecting continuity.

By considering the arc-length parameterisation it is possible to give an alternate definition for G^n continuity. Given curves $\mathbf{P}(t)$ and $\mathbf{Q}(u)$ they are G^n continuous at a if when re-parameterised by arc length they satisfy the C^n continuity conditions at a [28].

Using this definition, consider the case when two curves meet with G^0 continuity. Then $\mathbf{P}(a) = \mathbf{Q}(a)$ and thus G^0 continuity matches position. When differentiating w.r.t. arc length the first derivative corresponds to the unit tangent (2.6). Therefore G^1 continuity ensures $\mathbf{T}_\mathbf{P}(a) = \mathbf{T}_\mathbf{Q}(a)$, where $\mathbf{T}_i(a)$ represents the unit tangent vector at a for the curves $i = \mathbf{P}, \mathbf{Q}$. Thus G^1 continuity matches unit tangents (i.e. tangent direction). Differentiating again w.r.t. arc length gives the unit normal vector multiplied by the curvature (see section 2.6). Assuming G^1 continuity, the unit tangents and hence unit normal vectors coincide (given that they are oriented equivalently by condition 2.4(iii)). Therefore G^2 continuity ensures that $\kappa_\mathbf{P}(a)\mathbf{N}_\mathbf{P}(a) = \kappa_\mathbf{Q}(a)\mathbf{N}_\mathbf{Q}(a)$, where $\kappa_i(a)$ and $\mathbf{N}_i(a)$ represent the curvature and unit normal vector at a respectively for curves $i = \mathbf{P}, \mathbf{Q}$. Furthermore, $\mathbf{N}_\mathbf{P}(a) = \mathbf{N}_\mathbf{Q}(a)$ and thus G^2 continuity matches curvature values.

Each level of G^n continuity can be calculated by using the chain rule. The G^n continuity conditions for $n = 0 \ldots 3$ are given below as

$$
\begin{aligned}
n = 0 : \qquad & \mathbf{P}(a) = \mathbf{Q}(a), \\
n = 1 : \qquad & \frac{d\mathbf{P}(a)}{dt} = \frac{d\mathbf{Q}(a)}{dv}\frac{dv(a)}{du},
\end{aligned}
$$

$$
\begin{aligned}
n = 2: \qquad \frac{d^2\mathbf{P}(a)}{dt^2} &= \frac{d\mathbf{Q}(a)}{dv}\frac{d^2v(a)}{du^2} + \frac{d^2\mathbf{Q}(a)}{dv^2}\left(\frac{dv(a)}{du}\right)^2, \\
n = 3: \qquad \frac{d^3\mathbf{P}(a)}{dt^3} &= \frac{d\mathbf{Q}(a)}{dv}\frac{d^3v(a)}{du^3} + 3\frac{d^2\mathbf{Q}(a)}{dv^2}\frac{d^2\mathbf{Q}(a)}{dv^2}\frac{d^2v(a)}{du^2} + \frac{d^3\mathbf{Q}(a)}{dv^3}\left(\frac{dv(a)}{du}\right)^3.
\end{aligned}
$$

Further levels can be calculated by successive applications of the chain rule [28], or alternatively via Faà di Bruno's formula [29].

Letting $\alpha_i = \dfrac{d^i v(a)}{du^i}$, the G^n conditions can be expressed in matrix form. For example, the G^3 matrix is

$$
\begin{bmatrix} \mathbf{P}(a) \\ \mathbf{P}'(a) \\ \mathbf{P}''(a) \\ \mathbf{P}'''(a) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \alpha_1 & 0 & 0 \\ 0 & \alpha_2 & \alpha_1^2 & 0 \\ 0 & \alpha_3 & 3\alpha_1\alpha_2 & \alpha_1^3 \end{bmatrix} \begin{bmatrix} \mathbf{Q}(a) \\ \mathbf{Q}'(a) \\ \mathbf{Q}''(a) \\ \mathbf{Q}'''(a) \end{bmatrix}. \tag{2.13}
$$

The α_i's are referred to as the shape factors. They can be real, free variables with the only restriction that $\alpha_1 > 0$ to ensure (2.4(iii)).

Consider the scenario when $\alpha_1 = 1$ and $\alpha_i = 0$ for $i = 2 \ldots n$. Then the C^n parametric continuity conditions are satisfied and the curve corresponds to the Hermite interpolant.

2.9 Establishing properties for the approximation

When trying to approximate the GCS with a CAD compatible curve, non-rational polynomials are a suitable choice (see sections 1.1 and 2.4). Expressing the polynomial in Bézier form will allow easier manipulation of the curve by utilising the properties discussed in section 2.3.2.

Furthermore, a certain level of continuity should be achieved at the endpoints. This will prove important when joining segments together (see section 3.5). The curves should be made to match end points and tangent directions so that joints do not have gaps or kinks in them. An essential characteristic of the GCS, which the Bézier approximation should also possess, is that of a high quality curvature profile. As it is this feature to be approximated, interpolation of the end curvature values is desirable. Furthermore, when curve segments are joined, there will be no discontinuities in the overall curvature profile. Arbitrary precision of the relative curvature error metric, ϵ, will then be possible (see Chapter 4). These constraints are equivalent to the G^2 continuity conditions.

A further issue is what the degree of the Bézier should be. When trying to establish G^2 continuity at the end points, an independent control of the end curvatures will be a useful property. The curvature depends on the first and second derivatives of the curve via (2.10) so independent control of these are required. By the derivative control property of Bézier curves (see section 2.3.2) the smallest degree polynomial which allows this is a quintic.

2.10 Summary

The chapter began by looking at parametric planar curves. Then polynomial curves and the Bézier form were examined in detail. By looking at re-parameterisations it was possible to express a curve in terms of arc-length. This led to the definition of curvature and its various properties. This allowed for the definition of the GCS and highlighted its incompatibility with CAD. After looking at different continuities, the G^2 conditions for the Bézier were arrived at and the degree argued to be 5.

Relating back to the objective for the thesis, it is the intention of this research to find an acceptable G^2 quintic Bézier polynomial to approximate the GCS. The next section looks at the GCS in more detail to establish some important properties.

Chapter 3

General properties of the GCS

This chapter looks at the GCS in more detail. To begin the different types of curves that the GCSs encompass are derived. Approximation techniques to these degenerate curve types, as discussed in the literature review, are given to show their relation to a general GCS. The expression of the GCS is then modified to simplify further manipulation. The formula for the winding angle of a GCS is presented leading to the definition of a GCS class. Then the GCS is reformatted so that it is defined by a new set of parameters $\{\theta, u, t\}$ which are used later on in the G^{2+} method presented in Chapter 6. This chapter finishes by describing the process of GCS subdivision.

3.1 Degenerate forms of the GCS

The family of GCS curves accommodate several subsets of other recognised curve types. Recall the curvature equation of a GCS,

$$\kappa(s) = \frac{p + qs}{S + rs} \qquad s \in [0, S]. \tag{3.1}$$

- If $p = 0$ and $q = 0$ then $\kappa(s) = 0$ and the curve is a straight line [8].
- If $q = 0$ and $r = 0$ then $\kappa(s) = \frac{p}{S}$ and the curve is a circular arc [18].
- If $q \neq 0$ and $r = 0$ then $\kappa(s) = \frac{(p+q)s}{S}$ and the curve is a Cornu spiral [14].
- If $q = 0$ and $r \neq 0$ then $\kappa(s) = \frac{p}{(S+rs)}$ and the curve is a logarithmic spiral [19].

Existing approximation methods to these degenerate curve types were discussed in section 1.3. None of these techniques could be applied across every curve type. However, the insight gained from the literature review will help shape the eventual approximation method produced in this thesis. Elements of the existing methods which will prove useful include exploiting symmetrical properties, deriving geometrical constraints and developing robustness through curve subdivision.

3.2 An alternative expression for the GCS

By labelling the curvature at the start of the curve $\kappa(0) = \kappa_0$ and the curvature at the end of the curve to be $\kappa(S) = \kappa_1$ it can be shown that [14]

$$p = \kappa_0 S \qquad\qquad q = \kappa_1 - \kappa_0 + r\kappa_1.$$

Thus the GCS curvature equation of (3.1) can be rewritten as

$$\kappa(s) = \frac{\kappa_0 S + (\kappa_1 - \kappa_0 + r\kappa_1)s}{S + rs} \qquad\qquad s \in [0, S],$$

where S is the total arc length of the GCS and r is referred to as the shape factor.

3.2.1 The shape factor r

Given κ_0, κ_1 and S it is possible to manipulate the curvature profile and thus the shape of a curve by varying r. When $r = 0$ the curvature profile is linear. As r tends towards -1 this has the effect of pulling the curvature profile towards κ_1. Conversely, as $r \to \infty$ the curvature profile is pulled towards κ_0. Figure 3.1 illustrates the effect of varying r and the subsequent effect on the synthesised curves.

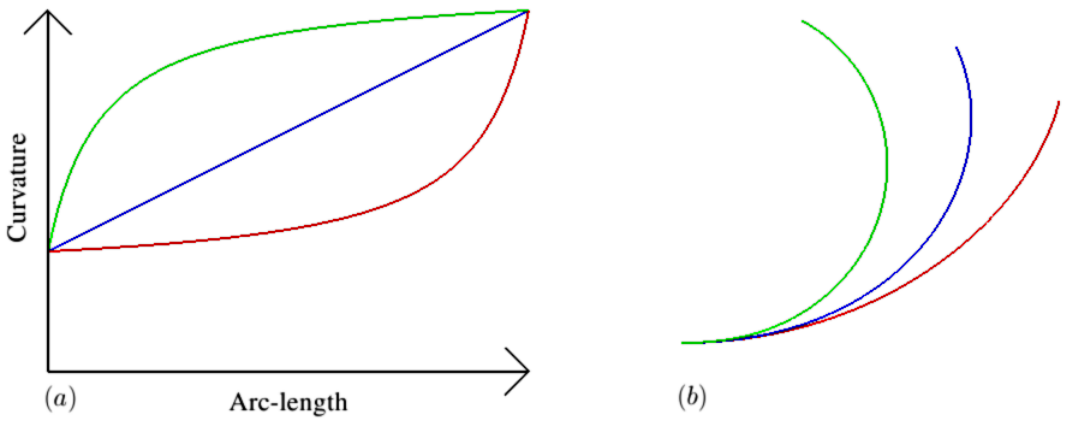

Figure 3.1: (a) Curvature profiles when $r = [-0.9, 0, 9]$. (b) Examples of synthesised curves corresponding to $r = [-0.9, 0, 9]$.

3.2.2 Normal form

The curvature profile of the GCS only describes the shape of the curve. Therefore, in order to completely define the curve, its initial location, (x_0, y_0), and orientation, θ_0, must be defined (2.12).

A scaling factor of $\alpha \in \mathbb{R}$ may also be applied to the curve (see section 2.2) resulting in new values for the parameters [15]:

$$S^* = \alpha S, \qquad \kappa_0^* = \frac{\kappa_0}{\alpha}, \qquad \kappa_1^* = \frac{\kappa_1}{\alpha}, \qquad r^* = r. \tag{3.2}$$

Thus, given a GCS, it is always possible to express it in *normal form*. This is achieved by firstly translating the GCS to the origin (i.e. setting $x_0 = 0, y_0 = 0$). The curve is then rotated so that the initial tangent is in the direction of the positive x-axis ($\theta_0 = 0$). Finally the curve is scaled to make the arc length equal to 1 ($S = 1$).

The curvature profile of a normalised GCS is therefore completely defined by the three parameters $\{\kappa_0, \kappa_1, r\}$. The original GCS on the other hand is defined by $\{\kappa_0, \kappa_1, r, S; x_0, y_0, \theta_0\}$. Thus dealing with a normalised GCS serves to reduce the degrees of freedom of a GCS.

To retrieve the original GCS from its normal form, the appropriate inverse transformations can be applied. It is possible to find an approximation to a general GCS by considering the approximation to the normalised GCS. This is because the transformations can be applied to the approximation. This is a valid technique for Bézier curves due to their affine invariance (see section 2.3.2). Thus for the remainder of the thesis it is assumed that a GCS is in normal form unless explicitly stated otherwise.

3.2.3 The modified shape factor u

The shape factor r of a GCS lies in the infinite domain $r \in (-1, \infty)$. For analytical purposes, it may be preferred to have the shape factor lie in a finite domain. This was the inspiration for creating the *modified shape factor* u. It was defined as the rational linear transformation of r so that the domain of u lies in $u \in (0, 1)$

$$u(r) = \frac{r+1}{r+2}, \quad r(u) = \frac{1-2u}{u-1}. \tag{3.3}$$

It is worth noting that this re-parameterisation of the shape factor does not affect the curvature

profile and thus the shape of the curve. This is verified by:

$$r(u(r)) = \frac{1 - 2\frac{r+1}{r+2}}{\frac{r+1}{r+2} - 1} = \frac{r + 2 - 2r - 2}{r + 1 - r - 2} = \frac{-r}{-1} = r.$$

Figure 3.2 illustrates the relationship of the modified shape factor u with the original shape factor r. The modified shape factor u will be used later on to develop the G^{2+} method in Chapter 6.

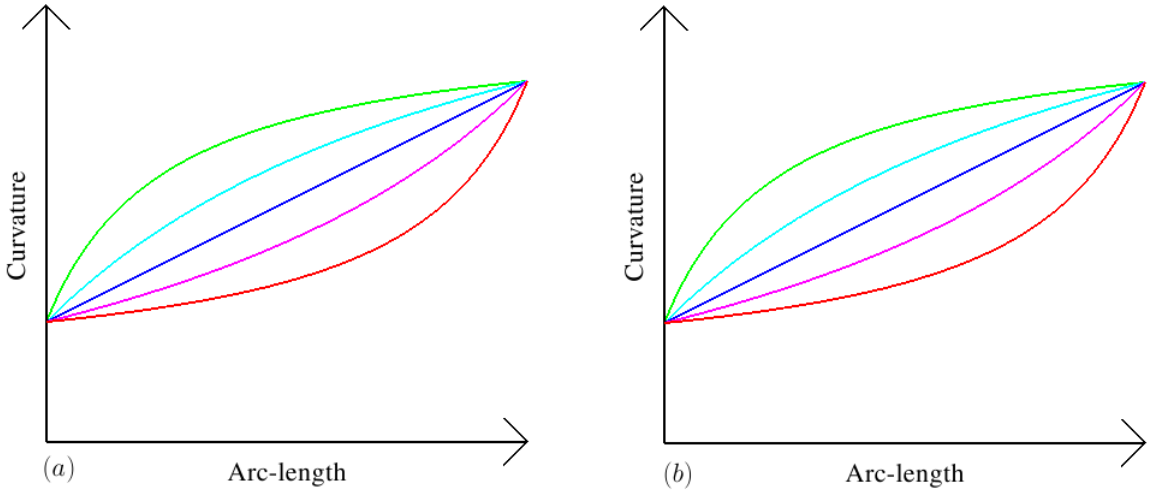

Figure 3.2: (a) A curvature profile when $r = [-0.8, -0.5, 0, 1, 4]$. (b) A curvature profile when $u = [0.17, 0.33, 0.5, 0.67, 0.83]$

3.3 The winding angle of the GCS

The winding angle of a curve, θ, is a measure of how much the tangent has turned. It may be calculated as the difference between the angles of the tangent at the start and end of the curve.

Recall from section 2.2 that $\kappa(s) = \theta'(s)$. Therefore the winding angle of a GCS is

$$\theta = \int_0^S \kappa(s) ds.$$

Thus the winding angle of a GCS (in normal form) is

$$\begin{aligned}\theta &= \int_0^1 \kappa(s) ds = \int_0^1 \frac{(\kappa_1 - \kappa_0 + r\kappa_1)s + \kappa_0}{rs + 1} ds \\ &= \begin{cases} \dfrac{\kappa_0 + \kappa_1}{2} & \text{if } r = 0, \\ \dfrac{r(\kappa_1(1+r) - \kappa_0) + (1+r)(\kappa_0 - \kappa_1)\ln(1+r)}{r^2} & \text{otherwise.} \end{cases}\end{aligned}$$

Rearranging this equation the winding angle, θ, can be shown to be a convex combination of the start and end curvatures κ_0 and κ_1:

$$\theta = \lambda\kappa_0 + (1 - \lambda)\kappa_1 \tag{3.4}$$

where $\lambda \in [0, 1]$ and is calculated from r as [14]:

$$\lambda(r) = \begin{cases} \dfrac{1}{2} & \text{if } r = 0, \\ \dfrac{(1+r)\ln(1+r) - r}{r^2} & \text{otherwise.} \end{cases}$$

The parameter λ is used in the creation of a GCS class.

3.4 A GCS class

In Chapter 6 the G^{2+} method is developed and begins by seeking insight from the G^3 method of Chapter 5. To gain a greater insight into the G^3 method a set of GCS curves that possess the same winding angle, θ, and the same modified shape factor, u, will prove useful. This inspired the development of the *GCS class*.

A GCS class is denoted by $\mathcal{C}_{u,\theta}$, with each member $\mathcal{C}_{u,\theta}(t)$, $t \in \mathbb{R}$, equal to the GCS corresponding to

$$\kappa_0 = \theta + (1 - \lambda(r))t$$

$$\kappa_1 = \theta - \lambda(r)t$$

$$r = \frac{1 - 2u}{u - 1}$$

where $\kappa_0, \kappa_1 \in \mathbb{R}$ and $r \in (-1, \infty)$ and thus $u \in (0, 1)$ and $\theta, t \in \mathbb{R}$.

It can easily be verified by substiting these values of $\{\kappa_0, \kappa_1, r\}$ into (3.4) and (3.3) that varying the parameter t will not affect the winding angle θ nor the modified shape factor u. Furthermore,

$$t = \kappa_0 - \kappa_1. \tag{3.5}$$

Thus alternate defining parameters for the GCS are $\{\theta, u, t\}$. These new defining parameters will be useful for the G^{2+} approximation in Chapter 6.

3.5 GCS subdivision

In section 2.7.3 it was noted that the GCS could not in general be expressed by a finite polynomial. Therefore whenever an approximation is formed with a G^2 quintic Bézier there will always be an error. Even if it were somehow possible to minimise this error, the best approximation may still be unsatisfactory. If this scenario were to occur then a possible remedy is to split the curve into a number of subsections and approximate each one independently. These approximations could then be joined, ensuring continuity, to form an approximation to the original GCS (see section 6.3.3). This process of splitting the curve is referred to as *GCS subdivision*.

The (normalised) GCS, where (arc-length) parameter s lies in $[0, 1]$, can be split into two subsegments, GCS1 and GCS2 at (normalised) length λ along the curve ($0 < \lambda < 1$). The subsegments GCS1 and GCS2 are therefore defined for $s \in [0, \lambda]$ and $s \in [\lambda, 1]$ respectively. The junction point where the curve is to be split has curvature value

$$\kappa(\lambda) = \frac{\kappa_0 + (\kappa_1 - \kappa_0 + r\kappa_1)\lambda}{r\lambda + 1} = \kappa_\lambda. \tag{3.6}$$

In order to represent GCS1 and GCS2 as separate GCS curves it is required to find their defining parameters such that each curvature profile is of the form

$$\tilde{\kappa}(s) = \frac{\hat{\kappa_0}\hat{S} + (\hat{\kappa_1} - \hat{\kappa_0} + \hat{r}\hat{\kappa_1})s}{\hat{S} + \hat{r}s} \qquad\qquad s \in [0, \hat{S}].$$

Here ($\hat{}$) represents the defining parameters, $\{\hat{\kappa_0}, \hat{\kappa_1}, \hat{r}, \hat{S}\}$, of a subdivided GCS. These subdivided GCS curves will no longer be in normal form. To aid the algebra the normalisation

process will be applied after their defining parameters are calculated.

The values of $\hat{\kappa_0}, \hat{\kappa_1}, \hat{S}$ can be deduced immediately from the subdivision process. The values of $(\hat{\kappa_0}, \hat{\kappa_1}, \hat{S})$ are $(\kappa_0, \kappa_\lambda, \lambda)$ respectively for GCS1 and $(\kappa_\lambda, \kappa_1, 1-\lambda)$ respectively for GCS2. The shape factors, $\hat{r}$, of GCS1 and GCS2 are not so immediately obvious. They can be found by looking at the curvature derivative (see section 2.7.2),

$$\kappa'(s) = \frac{S(\kappa_1 - \kappa_0)(1+r)}{(rs+S)^2}.$$

Thus at the junction point, for GCS1

$$\begin{aligned}
\hat{\kappa}'(\hat{S}) &= \kappa'(\lambda) \\
\frac{\hat{S}(\hat{\kappa_1} - \hat{\kappa_0})(1+\hat{r})}{(\hat{r}\hat{S}+\hat{S})^2} &= \frac{S(\kappa_1 - \kappa_0)(1+r)}{(r\lambda+S)^2} \\
\frac{\lambda(\kappa_\lambda - \kappa_0)(1+\hat{r})}{(\hat{r}\lambda+\lambda)^2} &= \frac{(\kappa_1 - \kappa_0)(1+r)}{(r\lambda+1)^2} \\
\frac{(\kappa_\lambda - \kappa_0)}{(1+\hat{r})\lambda} &= \frac{(\kappa_1 - \kappa_0)(1+r)}{(r\lambda+1)^2}
\end{aligned}$$

So,

$$\begin{aligned}
(1+\hat{r}) &= \frac{(\kappa_\lambda - \kappa_0)(r\lambda+1)^2}{\lambda(\kappa_1 - \kappa_0)(1+r)} \\
&= \frac{(\kappa_\lambda(r\lambda+1) - \kappa_0(r\lambda+1))\,(r\lambda+1)}{\lambda(\kappa_1 - \kappa_0)(1+r)} \\
&= \frac{(\kappa_0 + (\kappa_1 + \kappa_1 r - \kappa_0)\lambda - \kappa_0(r\lambda+1))\,(r\lambda+1)}{\lambda(\kappa_1 - \kappa_0)(1+r)} \\
&= \frac{\lambda(\kappa_1 - \kappa_0)(1+r)(r\lambda+1)}{\lambda(\kappa_1 - \kappa_0)(1+r)} \\
&= (1+r\lambda).
\end{aligned}$$

Thus

$$\hat{r} = \lambda r.$$

Similarly, for GCS2

$$\begin{aligned}
\hat{\kappa}'(0) &= \kappa'(\lambda) \\
\frac{(\hat{\kappa_1} - \hat{\kappa_0})(1+\hat{r})}{\hat{S}} &= \frac{S(\kappa_1 - \kappa_0)(1+r)}{(r\lambda + S)^2} \\
\frac{(\kappa_1 - \kappa_\lambda)(1+\hat{r})}{(1-\lambda)} &= \frac{(\kappa_1 - \kappa_0)(1+r)}{(r\lambda+1)^2}
\end{aligned}$$

$$\begin{aligned}
\Rightarrow (1+\hat{r}) &= \frac{(\kappa_1 - \kappa_0)(1+r)(1-\lambda)}{(r\lambda+1)^2(\kappa_1 - \kappa_\lambda)} \\
&= \frac{(\kappa_1 - \kappa_0)(1+r)(1-\lambda)}{(r\lambda+1)((r\lambda+1)\kappa_1 - (r\lambda+1)\kappa_\lambda))} \\
&= \frac{(\kappa_1 - \kappa_0)(1+r)(1-\lambda)}{(r\lambda+1)((r\lambda+1)\kappa_1 - (\kappa_0 + (\kappa_1 + \kappa_1 r - \kappa_0)\lambda))} \\
&= \frac{(\kappa_1 - \kappa_0)(1+r)(1-\lambda)}{(r\lambda+1)(\kappa_1 - \kappa_0)(1-\lambda)} \\
&= \frac{(1+r)}{(1+r\lambda)}.
\end{aligned}$$

Thus

$$\hat{r} = \frac{r(1-\lambda)}{1+r\lambda}.$$

The two subdivided curves, GCS1 and GCS2, can then be put into normal form by applying a scaling factor to ensure $\hat{S} = 1$ (3.2). The values for the parameters when a (normalised) GCS

is subdivided at length λ is therefore given in Table 3.1.

GCS	$\longmapsto$	GCS-1	GCS-2
κ_0	$\longmapsto$	$\kappa_0\lambda$	$\kappa_\lambda(1-\lambda)$
κ_1	$\longmapsto$	$\kappa_\lambda\lambda$	$\kappa_1(1-\lambda)$
r	$\longmapsto$	λr	$\dfrac{r(1-\lambda)}{1+r\lambda}$
$S=1$	$\longmapsto$	1	1

Table 3.1: Effect on the parameters $\{\kappa_0, \kappa_1, r\}$ when subdividing a GCS at $S = \lambda$.

3.6 Summary

In this chapter the GCS was studied in more detail. It began by highlighting important degenerate curve types of the GCS. Then an alternative expression for the GCS was presented in terms of initial and end curvatures, a shape factor and arc length corresponding to κ_0, κ_1, r and S respectively. The normal form for a GCS was then presented with a view to reduce the degrees of freedom for its definition. Details of how the shape factor r can be modified to lie in a finite domain were then presented.

After that the formula for the winding angle of a GCS was presented and was shown to be a convex combination of the start and end curvatures κ_0 and κ_1. This inspired the development of the GCS class, which resulted in a new spanning set for a GCS in terms of $\{\theta, u, t\}$. The chapter finished by looking at the process of subdivision, and the effect on the parameters is summarised in Table 3.1.

Chapter 4

The Error Metric

An error metric is used to measure the quality of an approximation. The metric is used to determine whether or not an approximation is satisfactory by comparing the error to some tolerance. Acceptable approximations yield errors less than the tolerance.

This chapter details the error metric used to assess the quality of a Bézier curve approximation to a GCS. In order to quantify the error some form of measurement for approximation accuracy must be employed. Deciding how this error is measured is discussed at the beginning of the chapter. After a suitable error metric has been defined it is then related to a tolerance to determine a condition for acceptable approximations.

4.1 Comparing curvature profiles

The motivation for using a GCS curve is that the curvature is a smooth monotonic rational linear function. This property, which ensures fairness, is what the approximation should be mimicking. Therefore when questioning how good an approximation is, a measure of error should reflect on its curvature.

This suggests that rather than using traditional measures of error, such as Hausdorff distance [30], the curvature profiles of the GCS and its approximation should be compared, as proposed in [23]. This involves re-parameterising the Bézier curve so that both Bézier and GCS curves are parameterised w.r.t. arc-length.

4.1.1 Re-parameterisation of the Bézier curve

The Bézier curve may be re-parameterised w.r.t. arc-length (see section 2.5.2). The curvature profile then takes values of $\kappa(s)$ for $s \in [0, S_b]$, where s represents arc-length and S_b is the total length of the Bézier curve. In order to compare curvature values the parameter s is normalised by a factor $\frac{1}{S_b}$ so that both curvature functions of the Bézier (κ_b) and the GCS (κ_g) curves range over the values $[0, 1]$ as proposed in [23]. Note this is equivalent to first scaling the Bézier curve so that its total arc-length is 1, and obtaining the curvature profile for this curve.

4.1.2 Absolute vs. relative error

Two ways to compare these curvature functions are the absolute curvature difference (ϵ_a) and the relative curvature difference (ϵ_r):

$$\epsilon_a(\kappa_b, \kappa_g; s) = |\kappa_b(s) - \kappa_g(s)|, \qquad \epsilon_r(\kappa_b, \kappa_g; s) = \frac{|\kappa_b(s) - \kappa_g(s)|}{|\kappa_g(s)|}.$$

Let us assume that an approximation has an absolute curvature error of ϵ. Scaling both the GCS and the approximation by λ the absolute curvature error between these two curves is:

$$\epsilon_r(\hat{\kappa_b}, \hat{\kappa_g}; s) = |\tfrac{1}{\lambda}\kappa_b(s) - \tfrac{1}{\lambda}\kappa_g(s)| = \tfrac{1}{\lambda}|\kappa_b(s) - \kappa_g(s)| = \tfrac{1}{\lambda}\epsilon.$$

Thus the error metric of absolute curvature difference depends upon a scaling factor. The normalisation of a GCS involves a scaling transformation and thus this measure will depend upon the arc length of the GCS before it was normalised.

Now let us assume that an approximation has a relative curvature error of ϵ. Scaling both the GCS and the approximation by λ the relative curvature error between these two curves is:

$$\epsilon_r(\hat{\kappa_b}, \hat{\kappa_g}; s) = \frac{|\frac{1}{\lambda}\kappa_b(s) - \frac{1}{\lambda}\kappa_g(s)|}{|\frac{1}{\lambda}\kappa_g(s)|} = \frac{|\kappa_b(s) - \kappa_g(s)|}{|\kappa_g(s)|} = \epsilon.$$

This shows that the error metric for relative curvature difference is invariant under scaling of the curve and thus more appealing.

However, this metric is not well defined when the curvature $\kappa_g(s)$ tends to 0. A point on the

curve when $\kappa(s) = 0$ is referred to as an *inflection point* and curves containing an inflection point are referred to as *inflecting curves*. Due to the monotonicity of the curvature of a GCS, there is at most one infection point. Divergence of the relative curvature error will occur local to this inflection point. Thus relative curvature error becomes less appropriate for smaller values of $|\kappa_g(s)|$ and incompatible when dealing with inflecting curves.

A more stable metric local to the inflection point is the absolute curvature difference. However, this metric does not possess the scaling invariance property used in the GCS normalisation process and is therefore less desirable. Thus when dealing with larger curvature values relative curvature error is preferred.

A compromise between these two metrics is found by taking the absolute error when the curvature of the GCS is small, i.e. $|\kappa_g(s)| < \mu$, and the relative error otherwise. The value of this tolerance, μ, will depend upon the units the curvature is measured in and can be seen as arbitrary. This is because the GCSs studied throughout this thesis are normalised (i.e. dimensionless) and thus a tolerance will depend upon the length of the original GCS segment. Applying a tolerance to the allowable length of a GCS will have an equivalent effect as to limiting the value of μ. By choosing $\mu = 1$ the metric can be realised as the minimum of the absolute and relative curvature differences. A consequence of this choice of μ ensures that the error metric achieves C^0 continuity as discussed in the following section.

4.2 The error metric ϵ

The error metric ϵ is defined as:

$$\begin{aligned}\epsilon(\kappa_b, \kappa_g; s) =& \min\{\epsilon_a(\kappa_b, \kappa_g; s), \epsilon_r(\kappa_b, \kappa_g; s)\} \\ =& \frac{|\kappa_b(s) - \kappa_g(s)|}{\max\{|\kappa_g(s)|, 1\}}.\end{aligned}$$

An approximation can thus be assigned an error value equal to the maximum error experienced throughout the domain of the function. That is:

$$\epsilon = \epsilon(\kappa_b, \kappa_g) = \max_{s\in[0,1]} \frac{|\kappa_b(s) - \kappa_g(s)|}{\max\{|\kappa_g(s)|, 1\}}. \tag{4.1}$$

Defining the error in this way ϵ is well defined in the sense that it does not diverge. Furthermore, as the error converges to zero the two curves coincide.

This can be shown by observing that:

$$\epsilon = 0 \qquad \Leftrightarrow \qquad \kappa_g(s) \equiv \kappa_b(s) \quad \forall s \in [0, 1].$$

This means that the GCS and the scaled Bézier curves (section 4.1.1) must have the same curvature profile. By the Fundamental Theory of Space Curves [15] they must be equivalent up to a rigid-body motion. Since the curves match the G^0 (position) conditions at the end points the GCS and the Bézier curve must have the same arc length and thus $S_b = 1$. This implies the

curves are identical.

Since the continuity of ϵ will be useful in Chapter 7 it is also noted that ϵ is C^0 continuous. Furthermore, C^1 continuity cannot be guaranteed if there exists $s_1, s_2 \in (0, 1)$ such that $|\kappa_g(s_1)| < 1$ and $|\kappa_g(s_2)| > 1$. A consequence of this is that the derivatives of ϵ are intractable.

4.2.1 Establishing a tolerance

An approximation can be considered acceptable if the error is within some predefined tolerance i.e. $\epsilon \leq \mu$. If μ is too large the level of accuracy diminishes. Conversely, if μ is too small then the tolerance cannot always be met. A reasonable suggestion that ensures high quality definition is $\mu = 0.05$ [14, 31]. Thus any curve that satisfies $\epsilon \leq 0.05$ is deemed an acceptable approximation.

4.3 Summary

In this chapter the error metric was developed to quantify how acceptable an approximation is. It was decided that the error should compare the curvature profiles of the GCS and the Bézier approximation. The Bézier curve was re-parameterised to enable direct comparison of curvature profiles.

The error was defined as the minimum of the absolute curvature difference and the absolute relative curvature difference. Using this error it is possible to classify approximations as satis-

factory, if $\epsilon \leq \mu$. A suitable tolerance was decided as $\mu = 0.05$ and thus approximations may be classified as acceptable if $\epsilon \leq 5\%$.

Chapter 5

The G^3 Method

The objective of this research is to find a suitable replacement for the GCS that will be CAD compatible. It was argued in section 2.9 that the form of the replacement should be a quintic Bézier curve interpolating the G^2 data at the endpoints. The chapter begins by deriving the formation of this G^2 quintic Bézier curve for approximation of a GCS. Further constraints are enforced leading to G^3 interpolation. This is used later on in the chapter to develop what is referred to as the G^3 *method* [32].

A G^3 interpolation and the geometrical meaning are explained. After solving these G^3 conditions, two free parameters remain. The geometrical significance of these free parameters relate to the end tangent magnitudes, which in turn suggest suitable values.

These initial values for the two free parameters are not always acceptable. This is a consequence of a divergence in the approximation. A solution to this problem involves varying the two

free parameters in a numerical search. Details of a search routine designed to overcome these shortcomings is then presented. The chapter finishes with a summary of the algorithm behind the G^3 method followed by concluding remarks on possible improvements.

5.1 A G^2 interpolation

In section 2.9 it was determined that the GCS approximation should be a quintic Bézier curve that satisfied the G^2 conditions. Thus the approximation should match the position, unit tangent and curvature values at the end points of the curve and seek to satisfy the equations (2.13)

$$\begin{bmatrix} \mathbf{V}(0) \\ \mathbf{V}'(0) \\ \mathbf{V}''(0) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \beta_1 & 0 \\ 0 & \beta_2 & \beta_1^2 \end{bmatrix} \begin{bmatrix} \mathbf{F}(0) \\ \mathbf{F}'(0) \\ \mathbf{F}''(0) \end{bmatrix} \tag{5.1}$$

and

$$\begin{bmatrix} \mathbf{V}(1) \\ \mathbf{V}'(1) \\ \mathbf{V}''(1) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \gamma_1 & 0 \\ 0 & \gamma_2 & \gamma_1^2 \end{bmatrix} \begin{bmatrix} \mathbf{F}(1) \\ \mathbf{F}'(1) \\ \mathbf{F}''(1) \end{bmatrix} \tag{5.2}$$

where β_1 and β_2 are the shape factors for the start point and γ_1 and γ_2 are the shape factors at the end point. End point derivative data for the quintic Bézier and the GCS must therefore be calculated.

5.1.1 Calculating the derivative data

The quintic Bézier

Derivatives of the quintic Bézier can be calculated from (2.2). The end point derivative data at both end points, up to the third derivatives are:

$$\begin{aligned}
\mathbf{V}(0) &= \mathbf{V}_0 & \mathbf{V}(1) &= \mathbf{V}_5 \\
\mathbf{V}'(0) &= 5(\mathbf{V}_1 - \mathbf{V}_0) & \mathbf{V}'(1) &= 5(\mathbf{V}_5 - \mathbf{V}_4) \\
\mathbf{V}''(0) &= 20(\mathbf{V}_2 - 2\mathbf{V}_1 + \mathbf{V}_0) & \mathbf{V}''(1) &= 20(\mathbf{V}_5 - 2\mathbf{V}_4 + \mathbf{V}_3) \\
\mathbf{V}'''(0) &= 60(\mathbf{V}_3 - 3\mathbf{V}_2 + 3\mathbf{V}_1 - \mathbf{V}_0) & \mathbf{V}'''(1) &= 60(\mathbf{V}_5 - 3\mathbf{V}_4 + 3\mathbf{V}_3 - \mathbf{V}_2)
\end{aligned}$$

The GCS

The parameterisation for a normalised GCS, $\mathbf{F}(t)$ for $t \in [0, 1]$, is given by (2.12)

$$\mathbf{F}(t) = \begin{bmatrix} \int_0^t \cos\left[\int_0^\sigma \kappa(s)ds\right]d\sigma \\ \int_0^t \sin\left[\int_0^\sigma \kappa(s)ds\right]d\sigma \end{bmatrix} = \begin{bmatrix} \int_0^t \cos\big(\theta(\sigma)\big)d\sigma \\ \int_0^t \sin\big(\theta(\sigma)\big)d\sigma \end{bmatrix}.$$

Thus

$$\mathbf{F}'(t) = \begin{bmatrix} \cos\big(\theta(t)\big) \\ \sin\big(\theta(t)\big) \end{bmatrix},$$

$$
\begin{aligned}
\mathbf{F}''(t) =& \kappa(t) \begin{bmatrix} -\sin\big(\theta(t)\big) \\ \cos\big(\theta(t)\big) \end{bmatrix}, \\
\mathbf{F}'''(t) =& \kappa'(t) \begin{bmatrix} -\sin\big(\theta(t)\big) \\ \cos\big(\theta(t)\big) \end{bmatrix} - \kappa(t)^2 \begin{bmatrix} \cos\big(\theta(t)\big) \\ \sin\big(\theta(t)\big) \end{bmatrix} \\
=& \mathbf{F}''(t)\frac{\kappa'(t)}{\kappa(t)} - \mathbf{F}'(t)\kappa(t)^2.
\end{aligned}
$$

The derivative data for a normalised GCS at the endpoints is therefore:

$$
\begin{aligned}
\mathbf{F}(0) =& (0,0) & \mathbf{F}(1) =& (x,y) \\
\mathbf{F}'(0) =& (1,0) & \mathbf{F}'(1) =& (\cos\theta, \sin\theta) \\
\mathbf{F}''(0) =& (0,\kappa_0) & \mathbf{F}''(1) =& \kappa_1(-\sin\theta, \cos\theta) \\
\mathbf{F}'''(0) =& \mathbf{F}''(0)\frac{\kappa'(0)}{\kappa(0)} - \mathbf{F}'(0)\kappa(0)^2 & \mathbf{F}'''(1) =& \mathbf{F}''(1)\frac{\kappa'(1)}{\kappa(1)} - \mathbf{F}'(1)\kappa(1)^2 \\
=& (-\kappa_0^2, (\kappa_1 - \kappa_0)(1+r)) & =& \mathbf{F}''(1)\frac{(\kappa_1 - \kappa_0)}{\kappa_1(1+r)} - \mathbf{F}'(1)\kappa_1^2
\end{aligned}
$$

where (x, y) is the end point and θ is the winding angle calculated from (3.4). The end points can be calculated by numerical integration of (2.12), such as Romberg integration [33].

5.1.2 A G^2 construction

The G^2 conditions at the start point are (5.1):

$$\begin{bmatrix} \mathbf{V}_0 \\ 5(\mathbf{V}_1 - \mathbf{V}_0) \\ 20(\mathbf{V}_2 - 2\mathbf{V}_1 + \mathbf{V}_0) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \beta_1 & 0 \\ 0 & \beta_2 & \beta_1^2 \end{bmatrix} \begin{bmatrix} \mathbf{F}(0) \\ \mathbf{F}'(0) \\ \mathbf{F}''(0) \end{bmatrix}.$$

thus

$$\begin{bmatrix} 1 & 0 & 0 \\ -5 & 5 & 0 \\ 20 & -40 & 20 \end{bmatrix} \begin{bmatrix} \mathbf{V}_0 \\ \mathbf{V}_1 \\ \mathbf{V}_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \beta_1 & 0 \\ 0 & \beta_2 & \beta_1^2 \end{bmatrix} \begin{bmatrix} \mathbf{F}(0) \\ \mathbf{F}'(0) \\ \mathbf{F}''(0) \end{bmatrix}$$

$$\begin{bmatrix} \mathbf{V}_0 \\ \mathbf{V}_1 \\ \mathbf{V}_2 \end{bmatrix} = \frac{1}{20} \begin{bmatrix} 20 & 0 & 0 \\ 20 & 4\beta_1 & 0 \\ 20 & \beta_2 + 8\beta_1 & \beta_1^2 \end{bmatrix} \begin{bmatrix} \mathbf{F}(0) \\ \mathbf{F}'(0) \\ \mathbf{F}''(0) \end{bmatrix}$$

Similary, the G^2 conditions at the end point are (5.2):

$$\begin{bmatrix} \mathbf{V}_5 \\ 5(\mathbf{V}_5 - \mathbf{V}_4) \\ 20(\mathbf{V}_5 - 2\mathbf{V}_4 + \mathbf{V}_3) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \gamma_1 & 0 \\ 0 & \gamma_2 & \gamma_1^2 \end{bmatrix} \begin{bmatrix} \mathbf{F}(1) \\ \mathbf{F}'(1) \\ \mathbf{F}''(1) \end{bmatrix}.$$

thus

$$\begin{bmatrix} 1 & 0 & 0 \\ 5 & -5 & 0 \\ 20 & -40 & 20 \end{bmatrix} \begin{bmatrix} \mathbf{V}_5 \\ \mathbf{V}_4 \\ \mathbf{V}_3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \gamma_1 & 0 \\ 0 & \gamma_2 & \gamma_1^2 \end{bmatrix} \begin{bmatrix} \mathbf{F}(1) \\ \mathbf{F}'(1) \\ \mathbf{F}''(1) \end{bmatrix}$$

$$\begin{bmatrix} \mathbf{V}_5 \\ \mathbf{V}_4 \\ \mathbf{V}_3 \end{bmatrix} = \frac{1}{20} \begin{bmatrix} 20 & 0 & 0 \\ 20 & -4\gamma_1 & 0 \\ 20 & \gamma_2 - 8\gamma_1 & \gamma_1^2 \end{bmatrix} \begin{bmatrix} \mathbf{F}(1) \\ \mathbf{F}'(1) \\ \mathbf{F}''(1) \end{bmatrix}.$$

The G^2 quintic Bézier can therefore be constructed by:

$$\begin{aligned} \mathbf{V}_0 &= \mathbf{F}(0) \\ \mathbf{V}_1 &= \mathbf{F}(0) + \frac{\beta_1}{5}\mathbf{F}'(0) \\ \mathbf{V}_2 &= \mathbf{F}(0) + (\frac{\beta_2}{20} + \frac{2\beta_1}{5})\mathbf{F}'(0) + \frac{\beta_1^2}{20}\mathbf{F}''(0) \\ \mathbf{V}_3 &= \mathbf{F}(1) + (\frac{\gamma_2}{20} - \frac{2\gamma_1}{5})\mathbf{F}'(1) + \frac{\gamma_1^2}{20}\mathbf{F}''(1) \\ \mathbf{V}_4 &= \mathbf{F}(1) + \frac{-\gamma_1}{5}\mathbf{F}'(1) \\ \mathbf{V}_5 &= \mathbf{F}(1). \end{aligned} \tag{5.3}$$

The G^2 construction has four degrees of freedom, namely the shape factors $\beta_1, \beta_2, \gamma_1, \gamma_2$. It is possible to use these free parameters to impose higher levels of continuity on the Bézier curve. This is the idea behind the G^3 interpolation.

5.2 A G^3 interpolation

The error of an approximation is directly related to its curvature profile. A G^2 interpolation matches the curvature values at the end points. Therefore when comparing the curvature profiles of the GCS and the Bézier the end points agree. To provide a better model of this curvature profile, the tangents of the curvature profile at the end points can also be equated. Figure 5.1 illustrates this idea by inspecting the curvature profile of a Hermite C^2 interpolant (section 2.8.2).

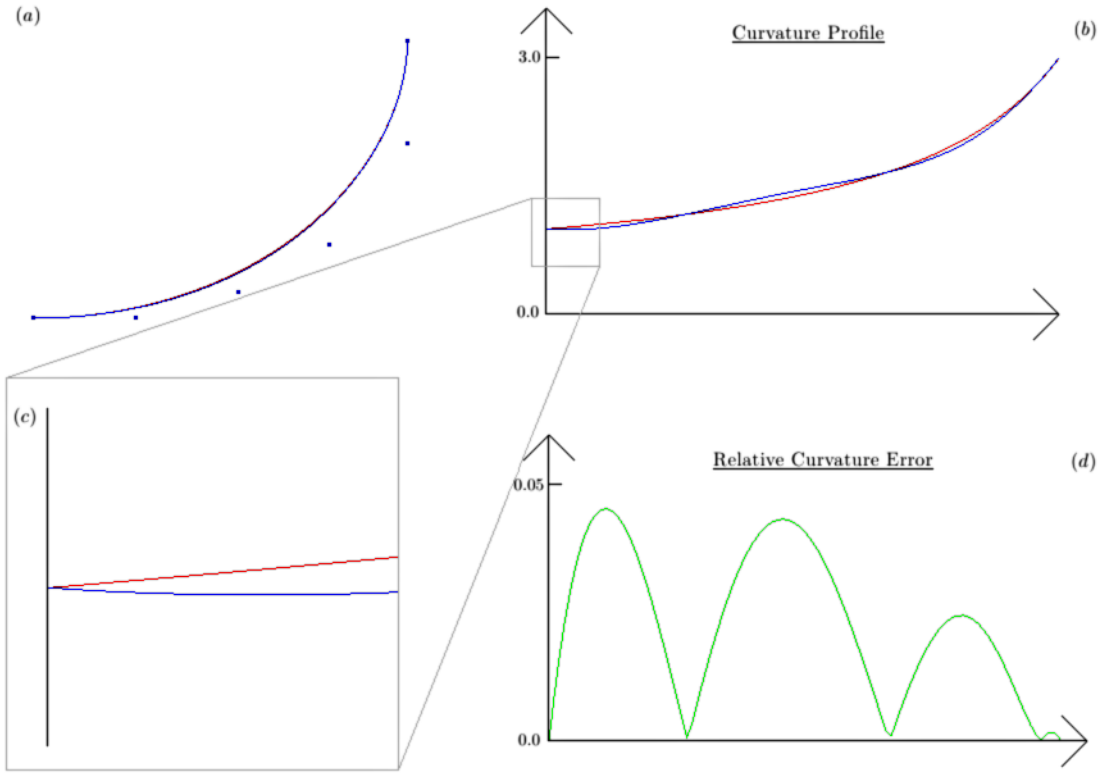

Figure 5.1: [a] A C^2 ($\beta_1 = \gamma_1 = 1, \beta_2 = \gamma_2 = 0$) Hermite approximation (blue) to the normalised GCS ($\kappa_0 = 1, \kappa_1 = 3, r = -0.75$) (red). [b] The corresponding curvature plot. [c] a close up of the initial curvature. [d] The relative curvature error (ϵ=0.045).

5.2.1 Geometric meaning

In order to match the curvature profile tangents at the end points, the following conditions need to be satisfied:

$$\begin{aligned} \frac{d}{ds}\kappa_{\mathbf{V}}(0) &= \frac{d}{ds}\kappa_{\mathbf{F}}(0) \\ \frac{d}{ds}\kappa_{\mathbf{V}}(1) &= \frac{d}{ds}\kappa_{\mathbf{F}}(1) \end{aligned} \tag{5.4}$$

where $\kappa_{\mathbf{V}}(u)$ and $\kappa_{\mathbf{F}}(t)$ are the curvature profiles of the Bézier and GCS curves respectively.

Since the curvature profile is just the second derivative w.r.t. arc length, (5.4) is equivalent to the G^3 continuity conditions from section 2.7.2 since:

$$\frac{d^3}{ds^3}\mathbf{V}(a) = \frac{d}{ds}\Big(\frac{d^2}{ds^2}\mathbf{V}(a)\Big) = \frac{d}{ds}\kappa_{\mathbf{V}}(a) = \frac{d}{ds}\kappa_{\mathbf{F}}(a) = \frac{d^3}{ds^3}\mathbf{F}(a).$$

5.2.2 Satisfying the G^3 conditions

Therefore in order to satisfy property (5.4) using a quintic Bézier to approximate a GCS, the following sets of equations can be formed via (2.13). To begin, the conditions at the start point

are:

$$\begin{bmatrix} \mathbf{V}_0 \\ 5(\mathbf{V}_1 - \mathbf{V}_0) \\ 20(\mathbf{V}_2 - 2\mathbf{V}_1 + \mathbf{V}_0) \\ 60(\mathbf{V}_3 - 3\mathbf{V}_2 + 3\mathbf{V}_1 - \mathbf{V}_0) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \beta_1 & 0 & 0 \\ 0 & \beta_2 & \beta_1^2 & 0 \\ 0 & \beta_3 & 3\beta_1\beta_2 & \beta_1^3 \end{bmatrix} \begin{bmatrix} \mathbf{F}(0) \\ \mathbf{F}'(0) \\ \mathbf{F}''(0) \\ \mathbf{F}'''(0) \end{bmatrix}$$

thus

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ -5 & 5 & 0 & 0 \\ 20 & -40 & 20 & 0 \\ -60 & 180 & -180 & 60 \end{bmatrix} \begin{bmatrix} \mathbf{V}_0 \\ \mathbf{V}_1 \\ \mathbf{V}_2 \\ \mathbf{V}_3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \beta_1 & 0 & 0 \\ 0 & \beta_2 & \beta_1^2 & 0 \\ 0 & \beta_3 & 3\beta_1\beta_2 & \beta_1^3 \end{bmatrix} \begin{bmatrix} \mathbf{F}(0) \\ \mathbf{F}'(0) \\ \mathbf{F}''(0) \\ \mathbf{F}'''(0) \end{bmatrix}$$

$$\begin{bmatrix} \mathbf{V}_0 \\ \mathbf{V}_1 \\ \mathbf{V}_2 \\ \mathbf{V}_3 \end{bmatrix} = \frac{1}{60} \begin{bmatrix} 60 & 0 & 0 & 0 \\ 60 & 12\beta_1 & 0 & 0 \\ 60 & 3\beta_2 + 24\beta_1 & 3\beta_1^2 & 0 \\ 60 & \beta_3 + 9\beta_2 + 36\beta_1 & 9\beta_1^2 + 3\beta_1\beta_2 & \beta_1^3 \end{bmatrix} \begin{bmatrix} \mathbf{F}(0) \\ \mathbf{F}'(0) \\ \mathbf{F}''(0) \\ \mathbf{F}'''(0) \end{bmatrix}.$$

Similarly, for the end point:

$$\begin{bmatrix} \mathbf{V}_5 \\ 5(\mathbf{V}_5 - \mathbf{V}_4) \\ 20(\mathbf{V}_5 - 2\mathbf{V}_4 + \mathbf{V}_3) \\ 60(\mathbf{V}_5 - 3\mathbf{V}_4 + 3\mathbf{V}_2 - \mathbf{V}_3) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \gamma_1 & 0 & 0 \\ 0 & \gamma_2 & \gamma_1^2 & 0 \\ 0 & \gamma_3 & 3\gamma_1\gamma_2 & \gamma_1^3 \end{bmatrix} \begin{bmatrix} \mathbf{F}(1) \\ \mathbf{F}'(1) \\ \mathbf{F}''(1) \\ \mathbf{F}'''(1) \end{bmatrix}$$

thus

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 5 & -5 & 0 & 0 \\ 20 & -40 & 20 & 0 \\ 60 & -180 & 180 & -60 \end{bmatrix} \begin{bmatrix} \mathbf{V}_5 \\ \mathbf{V}_4 \\ \mathbf{V}_3 \\ \mathbf{V}_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \gamma_1 & 0 & 0 \\ 0 & \gamma_2 & \gamma_1^2 & 0 \\ 0 & \gamma_3 & 3\gamma_1\gamma_2 & \gamma_1^3 \end{bmatrix} \begin{bmatrix} \mathbf{F}(1) \\ \mathbf{F}'(1) \\ \mathbf{F}''(1) \\ \mathbf{F}'''(1) \end{bmatrix}$$

$$\begin{bmatrix} \mathbf{V}_5 \\ \mathbf{V}_4 \\ \mathbf{V}_3 \\ \mathbf{V}_2 \end{bmatrix} = \frac{1}{60} \begin{bmatrix} 60 & 0 & 0 & 0 \\ 60 & -12\gamma_1 & 0 & 0 \\ 60 & 3\gamma_2 - 24\gamma_1 & 3\gamma_1^2 & 0 \\ 60 & -\gamma_3 + 9\gamma_2 - 36\gamma_1 & 9\gamma_1^2 - 3\gamma_1\gamma_2 & -\gamma_1^3 \end{bmatrix} \begin{bmatrix} \mathbf{F}(1) \\ \mathbf{F}'(1) \\ \mathbf{F}''(1) \\ \mathbf{F}'''(1) \end{bmatrix}.$$

Then $\mathbf{V}_0, \mathbf{V}_1, \mathbf{V}_4, \mathbf{V}_5$ are uniquely defined by the derivative data and the shape factors β_1 and γ_1. However, $\mathbf{V}_2$ and $\mathbf{V}_3$ need to simultaneously satisfy the constraints at the start and end points.

That is, the equations involving $\mathbf{V}_2$ and $\mathbf{V}_3$:

$$60\mathbf{F}(0) + (\beta_3 + 9\beta_2 + 36\beta_1)\mathbf{F}'(0) + (9\beta_1^2 + 3\beta_1\beta_2)\mathbf{F}''(0) + \beta_1^3\mathbf{F}'''(0) = 60\mathbf{F}(1) + (3\gamma_2 - 24\gamma_1)\mathbf{F}'(1) + 3\gamma_1^2\mathbf{F}''(1),$$

$$60\mathbf{F}(0) + (3\beta_2 + 24\beta_1)\mathbf{F}'(0) + 3\beta_1^2\mathbf{F}''(0) = 60\mathbf{F}(1) + (-\gamma_3 + 9\gamma_2 - 36\gamma_1)\mathbf{F}'(1) + (9\gamma_1^2 - 3\gamma_1\gamma_2)\mathbf{F}''(1) - \gamma_1^3\mathbf{F}'''(1), \tag{5.5}$$

need to be satisfied.

There are six degrees of freedom, namely the six shape parameters, and four equations to be satisfied, two each for the x and y values of $\mathbf{V}_2$ and $\mathbf{V}_3$. The four parameters $\beta_2, \beta_3, \gamma_2, \gamma_3$ behave linearly within these four equations hence a solution can be determined for these parameters given information about β_1 and γ_1. The details of the following solution can be found in Appendix 1. Solving for β_2 and γ_2 gives

$$\beta_2 = \frac{B(\beta_1, \gamma_1)}{D(\beta_1, \gamma_1)} \qquad \& \qquad \gamma_2 = \frac{G(\beta_1, \gamma_1)}{D(\beta_1, \gamma_1)} \tag{5.6}$$

where,

$$\begin{aligned} B(\beta_1, \gamma_1) =& \frac{1}{3}\big[-60x + 24\beta_1 - 24\cos(\theta)^2\beta_1(1+r) + 60\cos(\theta)^2 x(1+r) \\ &+ 3\cos(\theta)\kappa_1^2\gamma_1^3 - \beta_1^3\kappa_1^2\gamma_1 + 60y\kappa_1\gamma_1(1+r) \\ &+ 60\sin(\theta)\cos(\theta)y(1+r) - 3\kappa_0\sin(\theta)\cos(\theta)\beta_1^2(1+r) - 9\kappa_0\beta_1^2\kappa_1\gamma_1 \\ &- 2\beta_1^3\kappa_1^2\gamma_1 r - \beta_1^3\kappa_1^2\gamma_1 r^2 + \beta_1^3\kappa_1\gamma_1\kappa_0 + 24\beta_1 r \\ &- 60xr - 9\kappa_0\beta_1^2\kappa_1\gamma_1 r + 2\beta_1^3\kappa_1\gamma_1\kappa_0 r + \beta_1^3\kappa_1\gamma_1\kappa_0 r^2 \\ &+ 3\gamma_1^3\cos(\theta)\kappa_1^2 r - 15\sin(\theta)\kappa_1\gamma_1^2(1+r) - \sin(\theta)\gamma_1^3(\kappa_1 - \kappa_0) \\ &+ 9\cos^2(\theta)\sin(\theta)\kappa_1\gamma_1(1-\gamma_1)(1+r)\big]/(1+r). \end{aligned}$$

$$
\begin{aligned}
G(\beta_1,\gamma_1) =& \frac{1}{3}\big[-24\gamma_1(1+r) + 60\sin(\theta)y(1+r) - 3\cos(\theta)\kappa_0^2\beta_1^3(1+r) \\
&+ 15\sin(\theta)\kappa_0\beta_1^2(1+r) + 60\cos(\theta)y\beta_1\kappa_0(1+r) - 60x\sin(\theta)\beta_1\kappa_0(1+r) \\
&+ 3\sin(\theta)\cos(\theta)\kappa_1\gamma_1^2(1+r) + \sin(\theta)\beta_1^3\kappa_0 - \sin(\theta)\beta_1^3\kappa_1 \\
&+ \gamma_1^3\beta_1\kappa_0^2 - 2\sin(\theta)\beta_1^3\kappa_1 r - \sin(\theta)\beta_1^3\kappa_1 r^2 \\
&+ 2\sin(\theta)\beta_1^3\kappa_0 r + \sin(\theta)\beta_1^3\kappa_0 r^2 - \gamma_1^3\beta_1\kappa_0\kappa_1 \\
&+ 9\kappa_1\gamma_1^2\beta_1\kappa_0(1+r) + 24\cos(\theta)^2\gamma_1(1+r) \\
&+ 9\cos^2(\theta)\kappa_0\kappa_1\beta_1\gamma_1(1-\gamma_1)(1+r)\big]/(1+r).
\end{aligned}
$$

and

$$
D(\beta_1,\gamma_1) = \beta_1\gamma_1\kappa_0\kappa_1 - \sin^2(\theta). \tag{5.7}
$$

Therefore, given (β_1, γ_1) the G^3 interpolation can be found from (5.3). Although β_3 and γ_3 are not required for the construction of the Bézier, they can be used as a check to validate the G^3 conditions of (5.5). Their derivation is also shown in Appendix 1. The notation of $G^3(b, g)$ is hereby adopted to signify a G^3 interpolation with $\beta_1 = b$ and $\gamma_1 = g$.

5.2.3 The free parameters (β_1, γ_1)

After ensuring that the quintic Bézier curve matches the end curvature derivatives, equivalent to the G^3 conditions, there still remains two degrees of freedom. Insight into these free parameters can be gained from the G^1 conditions:

$$\mathbf{V}'(0) = \beta_1 \mathbf{F}'(0), \qquad \mathbf{V}'(1) = \gamma_1 \mathbf{F}'(1).$$

Therefore β_1 and γ_1 relate to the magnitude of the initial and end tangents. The size of the tangents will agree when $\beta_1 = \gamma_1 = 1$ as is the case for a C^1 interpolation (see section 2.8.2). Therefore sensible initial values for these shape factors are $\beta_1 = \gamma_1 = 1$.

5.2.4 Initial approximation

Figure 5.2 compares two examples of the $G^3(1,1)$ approximation to GCSs of similar shape. The GCS and Bézier curves are given along with the curvature profiles and relative curvature errors.

From Figure 5.2 it is apparent that this approximation method will not always suffice. The approximation on the left is acceptable but the one on the right is not because $\epsilon > 0.05$. (For more analysis on the performance of the $G^3(1,1)$ approximation see section 8.3.) Since the two curves are of similar shape the variation of accuracy suggests an instabilty in the construction method. The approximation on the right might be improved with a better position for the two internal vertices $\mathbf{V}_2$ and $\mathbf{V}_3$.

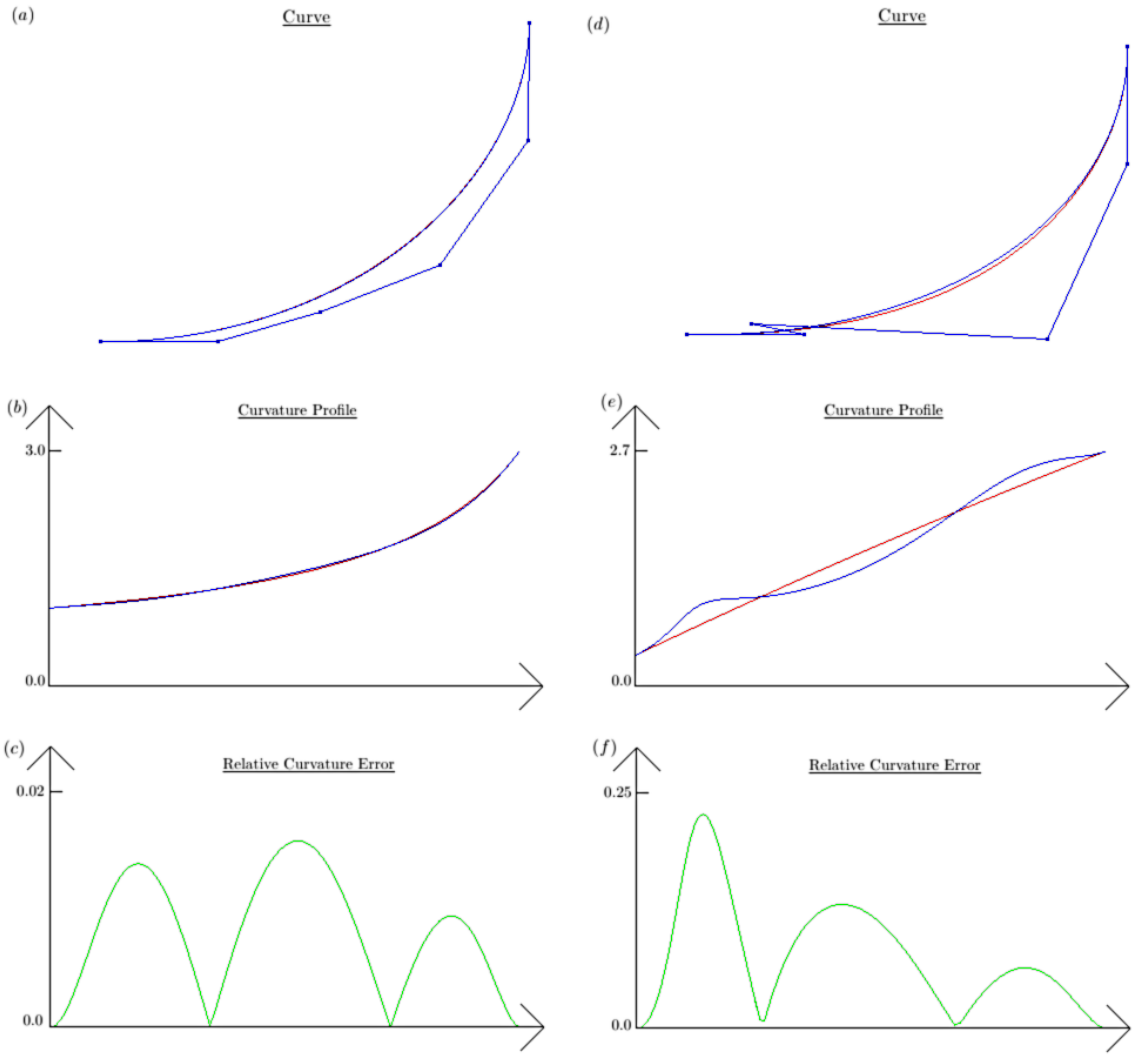

Figure 5.2: [a, d] $G^3(1,1)$ approximation (red) to the normalised GCS (red) ([a] - $\kappa_0 = 1, \kappa_1 = 3, r = -0.75$; [d] - $\kappa_0 = 0.36, \kappa_1 = 2.7, r = 0.1$). [b, e] The corresponding curvature plots. [c,f] The relative curvature errors ([c] - ϵ=0.016; [f] - ϵ=0.23).

5.2.5 Approximation divergence

From the construction method (5.3) it is apparent that the two internal vertices are affected by the second shape factors β_2, γ_2 whereas the the four external vertices are not. This suggests a

problem may lie with the second shape factors

$$\beta_2 = \frac{B(\beta_1, \gamma_1)}{D(\beta_1, \gamma_1)} \qquad \& \qquad \gamma_2 = \frac{G(\beta_1, \gamma_1)}{D(\beta_1, \gamma_1)}.$$

An observation is that as the denominator, $D(\beta_1, \gamma_1)$, tends to 0, a divergence of the values $\beta_2(\beta_1, \gamma_1)$, $\gamma_2(\beta_1, \gamma_1)$ towards ∞ occurs. This causes the approximation to become unreasonable. This behaviour will occur when the denominator is close to 0.

If the initial approximation, $G^3(1, 1)$, is unacceptable, possibly because of divergence, then a better solution might be found by considering different values of β_1, γ_1. These values can be determined via a numerical search on the two degrees of freedom.

There are numerous possible searching algorithms that could be used. However, properties of the underlying function means that specific algorithms are more suitable than others. Furthermore, an understanding of the search domain can be utilised to improve the performance of a search.

5.2.6 Outline of a search routine

A search routine designed specifically for the G^3 interpolant is presented here. The parameters (β_1, γ_1) are measured with respect to the error function ϵ which is to be minimised. The search is then performed until epsilon is within the desired tolerance $\epsilon \leq 0.05$.

When employing a search algorithm an initial starting point is often required [33]. Establishing

an appropriate starting point is essential to increase the efficiency of the rate of convergence to a solution and the likelihood of yielding one. This implies a search should ideally begin close to a solution and in a stable region.

A criterion, $\mathcal{P}$, for a possible starting point can be defined with the aim of increasing the efficiency of the search. Properties that the starting point should possess are first established. These properties identify restrictions on possible initial values, defining the criterion $\mathcal{P}$.

Recall that β_1 and γ_1 reflect the size of the initial and end tangent vectors. Obviously these values should not be negative, otherwise the tangents will point in the opposite direction (see section 2.8.2). Thus a lower bound of $\beta_1, \gamma_1 > 0$ is determined.

Also, β_1 and γ_1 should not be too large as this would result in $\mathbf{V}_0$ ($\mathbf{V}_5$) being too far from $\mathbf{V}_1$ ($\mathbf{V}_4$). Therefore an upper bound for these values should also be defined.

Recall that the arc-length of the curve is $S_b = \int_0^1 ||\dot{\mathbf{V}}(\tau)||d\tau \approx 1$. Considering the simplest Romberg approximation (equivalent to the trapezoid rule) [33] then

$$\begin{aligned} S &\approx \frac{1}{4}||\dot{\mathbf{V}}(0)|| + \frac{1}{2}||\dot{\mathbf{V}}(\frac{1}{2})|| + \frac{1}{4}||\dot{\mathbf{V}}(1)|| \\ &= \frac{1}{4}\beta_1 + \frac{1}{2}||\dot{\mathbf{V}}(\frac{1}{2})|| + \frac{1}{4}\gamma_1 \\ &> \frac{1}{4}(\beta_1 + \gamma_1). \end{aligned}$$

Therefore, a suitable upper bound is $\beta_1, \gamma_1 < 4$.

To avoid the function tending to infinity locally, the initial value (β_1, γ_1) is restricted such that

the denominator, $D(\beta_1, \gamma_1) > 0.1$. This was observed as approximately the smallest value of $D(\beta_1, \gamma_1)$ which still produces acceptable approximations when expected and therefore considered a sufficient distance away way from the divergent region. This value of 0.1 was arrived at empirically using data corresponding to the bounds discussed in section 6.3.1.

Therefore the criterion, $\mathcal{P}$, can be defined as:

$$\mathcal{P} = \{(\beta_1, \gamma_1) \in [0, 4] \times [0, 4] : |D(\beta_1, \gamma_1)| > 0.1\}.$$

Any value in $\mathcal{P}$ could be used to start a search, however better starting points can be argued for.

Previously, it was noted that large values for $\beta_2(\beta_1, \gamma_1)$ and $\gamma_2(\beta_1, \gamma_1)$ contributed to a poor approximation. Therefore an idea might be to control these values.

Again drawing inspiration from the C^2 Hermite model, consider $\beta_2 = \gamma_2 = 0$. This implies that the first and second parametric derivatives are orthogonal, a property that agrees with the GCS [31]. Therefore determining values which yield $\beta_2(\beta_1, \gamma_1) = \gamma_2(\beta_1, \gamma_1) = 0$ may provide a good starting value.

To calculate which values of β_1 and γ_1 give $\beta_2 = \gamma_2 = 0$, a set of equations must be solved, equivalent to $B(\beta_1, \gamma_1) = G(\beta_1, \gamma_1) = 0$ from (5.6). These equations can be realised as the intersection of two bivariate polynomials of degree four and thus have at most 16 solutions by Bézout's Theorem [34].

These solutions must also satisfy $\mathcal{P}$ in order to be a suitable initial point. If there are multiple suitable solutions then the one with the smallest error should be used. In the event no such

solutions exist, a random value in $\mathcal{P}$ can be chosen as the starting point for the search.

A search may not be necessary if this initial value already meets the tolerance. If this is not the case a search in 2-dimensions can now be implemented. Derivative data of the ϵ function is intractable (see section 4.2) and so a suitable numerical minimisation routine is Powell's method [33].

The method begins from an initial point and minimises along a 1-dimensional line of the 2-dimensional domain. The direction of the line is then adaptively altered after every iteration [33] and a new minimum sought. This minimum is guaranteed to be at most the size of the previous minimum. A suitable initial 1-dimensional line of the 2-dimensional search domain is $(\beta_1, \gamma_1) = (1, -1)$ which maximises the difference between β_1 and γ_1. The process is repeated until either a desired tolerance, $\epsilon \leq \mu$, is achieved or the the search fails.

This can occur because either the number of iterations exceeds a preset number or a local minima has been discovered such that $\epsilon > \mu$. If this scenario were to occur then two alternative options are to either increase the tolerance, μ, or split the curve into smaller segments.

5.3 Algorithm summary of the G^3 method

Given (β_1, γ_1), the G^3 method defines the control vertices of the quintic Bézier curve as

$$
\begin{aligned}
\mathbf{V}_0 &= \begin{pmatrix} 0 \\ 0 \end{pmatrix} \\
\mathbf{V}_1 &= \frac{\beta_1}{5}\begin{pmatrix} 1 \\ 0 \end{pmatrix} \\
\mathbf{V}_2 &= \left(\frac{\beta_2(\beta_1,\gamma_1)}{20} + \frac{2\beta_1}{5}\right)\begin{pmatrix} 1 \\ 0 \end{pmatrix} + \frac{\beta_1^2}{20}\begin{pmatrix} 0 \\ \kappa_0 \end{pmatrix} \\
\mathbf{V}_3 &= \begin{pmatrix} x \\ y \end{pmatrix} + \left(\frac{\gamma_2(\beta_1,\gamma_1)}{20} - \frac{2\gamma_1}{5}\right)\begin{pmatrix} \cos\theta \\ \sin\theta \end{pmatrix} + \frac{\gamma_1^2}{20}\kappa_1\begin{pmatrix} -\sin\theta \\ \cos\theta \end{pmatrix} \\
\mathbf{V}_4 &= \begin{pmatrix} x \\ y \end{pmatrix} - \frac{\gamma_1}{5}\begin{pmatrix} \cos\theta \\ \sin\theta \end{pmatrix} \\
\mathbf{V}_5 &= \begin{pmatrix} x \\ y \end{pmatrix}.
\end{aligned}
$$

where $\beta_2(\beta_1,\gamma_1), \gamma_2(\beta_1,\gamma_1)$ are calculated from (5.6).

A short summary of the construction algorithm follows. As soon as an acceptable approximation is found or the number of iterations exceeds a preset number the algorithm terminates.

Step 1. Apply the initial G^3 approximation with $\beta_1 = \gamma_1 = 1$.

If $\epsilon \leq \mu$ then the approximation is acceptable.

Otherwise a search is required.

Step 2. Find an initial (β_1, γ_1) for the search. Begin by solving $\beta_2 = \gamma_2 = 0$.

Step 3. Pick the solution with smallest ϵ that also satisfies $\mathcal{P}$.

If none exist then pick a random value in $\mathcal{P}$.

Step 4. Apply Powell's method from the starting value,

re-iterating until either an acceptable value is found

or search fails.

Step 5. If Step 4 fails either split the curve or increase the tolerance, μ.

Then start the approximation again.

5.4 Summary

This chapter developed the G^3 method, an approximation to the GCS satisfying the G^3 conditions at the end points. The construction used a quintic Bézier curve leaving two degrees of freedom in the form of the shape factors (β_1, γ_1). A geometric interpretation of these shape factors suggested a value of $\beta_1 = \gamma_1 = 1$. Using these values an initial approximation could be formed.

However, this initial choice for (β_1, γ_1) was not always suitable. The second shape factors (β_2, γ_2) share a common denominator, $D(\beta_1, \gamma_1)$ (5.6), and divergence of these shape factors

occurred as $D(\beta_1, \gamma_1) \to 0$. This had the effect of moving the two internal control vertices $\mathbf{V}_2$ and $\mathbf{V}_3$ towards $\pm\infty$.

To overcome this scenario, whenever the initial approximation $G^3(1, 1)$ did not provide a satisfactory approximation an alternative set of values for (β_1, γ_1) was determined via a numerical search. Details how the search routine should be implemented were given. Finally, should a satisfactory approximation not be available then either the tolerance could be increased, or the curve could be subdivided into smaller sections. This completed the description of the G^3 method.

The purpose of creating the G^3 method was to replace the GCS with a suitable CAD compatible quintic Bézier curve. However, due to the nature of the method, there are still some issues with CAD compatibility. Two concerns come about as a result of using a numerical search.

Firstly, it is difficult to measure the efficiency of the search routine. A potential problem is that if it is too computationally expensive to construct the Bézier then it will not be suitable for practical use. Secondly, there is no guarantee that a search will be able to find a solution.

An improvement to the G^3 method, which removes the requirement of a numerical search, is required to satisfy compatibility with CAD. The approximation construction presented in the next chapter, the G^{2+} method, resolves this issue. Examples of applications of the G^3 method and a discussion on the method's effectiveness can be found in Chapter 8.

Chapter 6

The G^{2+} Method

The previous chapter presented the G^3 method where, given values for the shape factors (β_1, γ_1), an approximation to the GCS could be formed. A shortcoming of the method was a divergence in the approximation. The solution to this problem involved varying the two free parameters in a numerical search. This process is computationally expensive and cannot guarantee a solution and so is not immediately suitable for direct CAD implementation.

This chapter presents a new approximation method to the GCS called the G^{2+} *method* [35] which addresses these issues. To begin, insight of the shape factors (β_1, γ_1) from Chapter 5 is gained by observing results from a G^3 *optimal search*. Next, the issue of divergent approximations is addressed and observations suggest a different approach for the definition of the second shape factors (β_2, γ_2).

In order to guarantee a satisfactory approximation, restrictions on the type of GCS are imposed.

These correspond to limiting the winding angle θ, the change in curvature t, and the modified shape factor u to lie within identified bounds. If a GCS does not meet these bounds then a process of subdividing the curve until subsequent subdivisions satisfy these bounds is outlined. This approach was not appropriate for the G^3 method since the divergent (non-C^0) properties meant that bounds could not be verified (see section 7.1.3). The chapter finishes with a summary of the G^{2+} method.

The G^{2+} method improves on the G^3 method because it does not involve a numerical search. Furthermore, if the GCS lies within certain boundary conditions an acceptable approximation can always be found. This claim is dealt with in Chapter 7.

6.1 A G^3 optimal search

From section 5.2.5, a shortcoming of the G^3 method was identified as a divergence in the shape factors β_2, γ_2. This corresponded to a zero in their denominator identified from (5.7):

$$D(\beta_1, \gamma_1) = \beta_1\gamma_1\kappa_0\kappa_1 - \sin^2(\theta).$$

The zeros of (5.7), for a given GCS, depend upon the values chosen for the first shape factors. Establishing an expression for suitable values of (β_1, γ_1), for a given GCS, will uniquely determine this value (5.7). It will also eliminate the need for determination via a search. Further insight into these shape factors, β_1 and γ_1, will help to establish properties for appropriate values.

Suitable unique values of (β_1, γ_1) for each GCS can be determined by performing a search as outlined in section 5.11 until a local minimum is found. Although this may not be the global minimum and thus the true optimum value, the values will provide insight to help determine a relationship with the GCS.

Results of these numerical searches are given in Figures 6.1-6.3. Figure 6.1 examines the values of the error ϵ resulting from the numerical search. Figures 6.2 and 6.3 detail the derived values for the shape factors β_1 and γ_1 respectively.

Each figure looks at a variety of GCS curves. The graphs on the left hand side of each figure correspond to the GCS curves that have a winding angle of zero ($\theta = 0$). The graphs on the right hand side have a winding angle of 90 degrees ($\theta = \frac{\pi}{2}$). The x-axes of the graphs correspond to a variation of the modified shape factor u with $u \in (0, 1)$. The y-axes of the graphs correspond to the change in curvature t with $t \in (0, \pi)$. These values correspond to bounds which will be discussed in section 6.3.1 with the exception to values of u. These have been chosen to highlight divergent properties as $u \to 0, 1$. Each GCS is assigned a colour from the hot-cold scale that is given to the right of the graphs. The (x, y) values were determined with a resolution of 100×100.

6.1.1 Observations

Figure 6.1 displays the curvature error (ϵ) of the approximations found from the optimised search. The values of ϵ on the left hand graph (where $\theta = 0$) range from 0 to 0.001 and from 0 to 0.01 on the right hand graph (where $\theta = \frac{\pi}{2}$). With exception to the few 'hotter' isolated

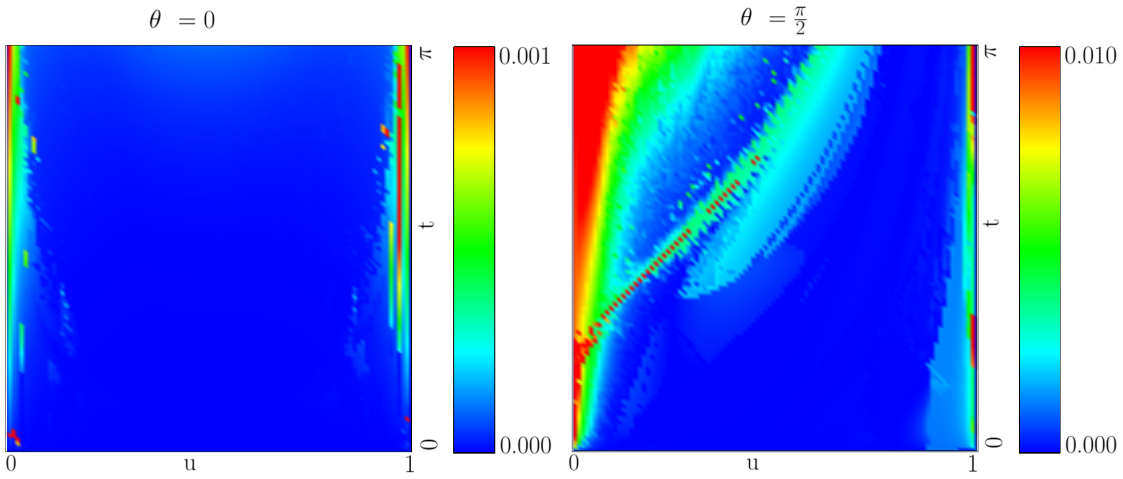

Figure 6.1: ϵ values from a G^3 search: $u \in (0,1), t \in [0,\pi]$ for $\theta = 0$ (left) and $\theta = \frac{\pi}{2}$ (right).

regions, as $u \to 0, 1$ (and as $D(\beta_1, \gamma_1) \to 0$ on the right hand graph), it is clear that a G^3 search can produce acceptable approximations.

When $\theta = \frac{\pi}{2}$ the search is less stable and produces poorer approximations. This suggests that as θ increases it becomes increasingly difficult to approximate, as expected intuitively. Another observation to note is that as $u \to 0, 1$ the quality of the approximations deteriorates as a consequence of the divergent behaviour at $u = 0, 1$. This suggests that a bound on u is necessary to guarantee acceptable approximations. Within certain regions of the graph the variations of errors are noticeably more erratic, giving the appearance of '*noisy data*'. This is attributed to the instability of the numerical approach coupled with divergent behaviour as $u \to 0, 1$ and $D(\beta_1, \gamma_1) \to 0$.

Figures 6.2 and 6.3 provide an insight into the shape factors β_1 and γ_1 respectively. Firstly, notice that as t varies, i.e. the vertical segments of the graph, the shape factors do not vary too much (within the stable regions). This suggests that the shape factors may be chosen independent of t. Furthermore, ignoring the effect of the noisy data, the adjacent graphs in Figures

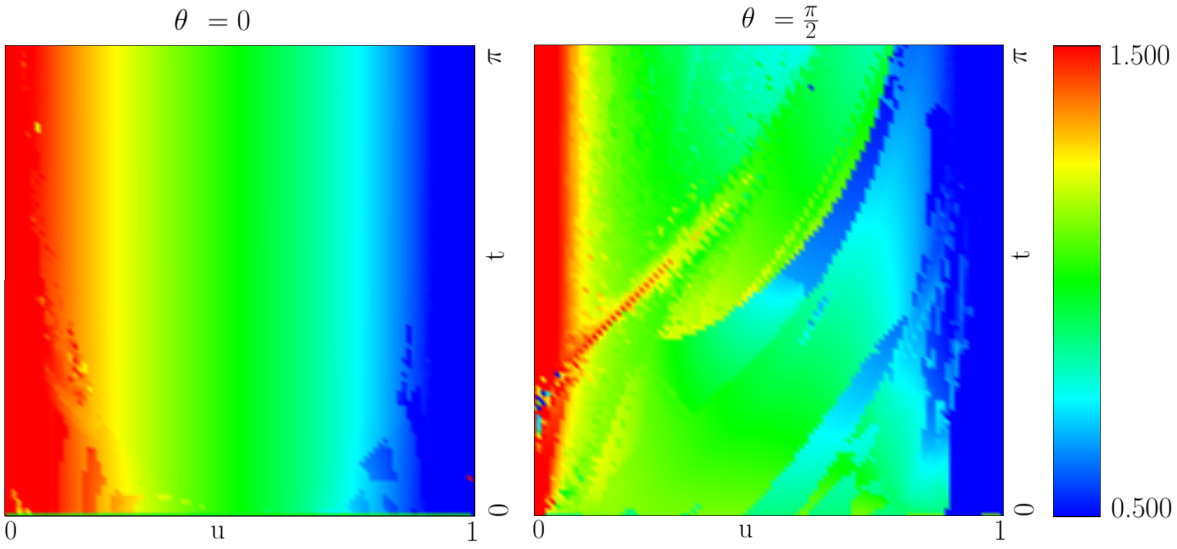

Figure 6.2: β_1 values from a G^3 search: $u \in (0, 1), t \in [0, \pi]$ for $\theta = 0$ (left) and $\theta = \frac{\pi}{2}$ (right).

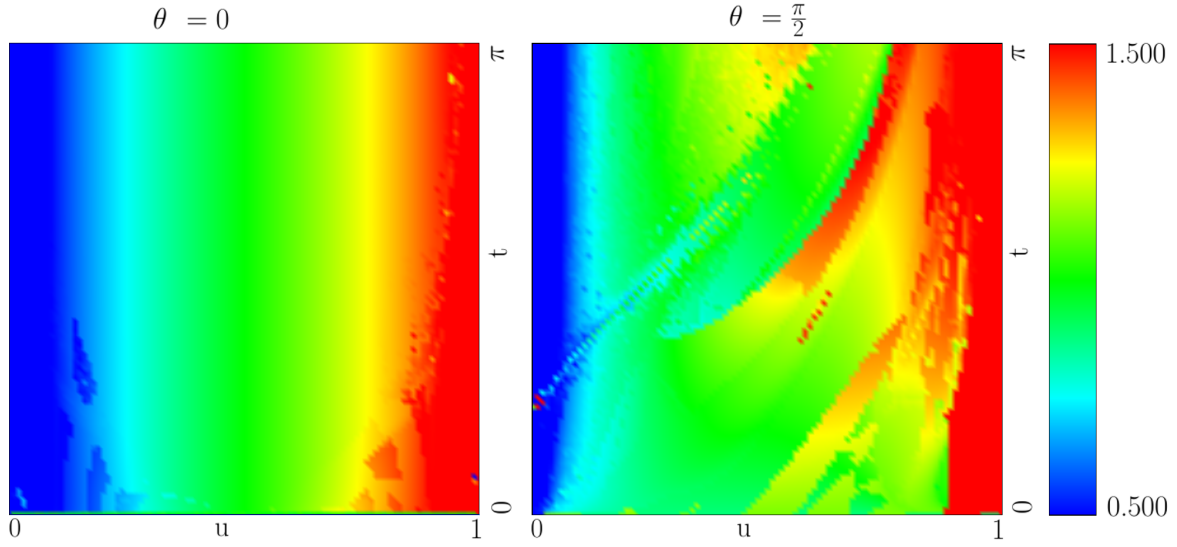

Figure 6.3: γ_1 values from a G^3 search: $u \in (0, 1), t \in [0, \pi]$ for $\theta = 0$ (left) and $\theta = \frac{\pi}{2}$ (right).

6.2 and 6.3 closely resemble each other. Hence an assertion is made that β_1 and γ_1 should be independent of θ. This will help simplify later algebra.

Secondly, notice how the shape parameters β_1 and γ_1 vary with the modified shape factor u. From Figure 6.2, β_2 varies from approximately 1.5 to 0.5 as u varies from 0 to 1. There is a

similar distribution for γ_2 from Figure 6.3, where the graphs appear alike after reflection about $u = 0.5$. This distribution appears to be linear.

6.1.2 Assigning values to the shape factors (β_1, γ_1)

A summary of the insight gathered in the previous section is that the shape factors (β_1, γ_1) may be considered independent of θ and t and linear in u. This relationship can be approximated as:

$$\beta_1 = 1.5 - u, \quad \gamma_1 = 0.5 + u. \tag{6.1}$$

G^3 approximations could then be performed given the newly defined values for (β_1, γ_1). When the denominator $D(\beta_1, \gamma_1)$ approaches zero, and hence the second shape factors (β_2, γ_2) diverge, poor approximations would still occur.

6.2 Divergence issues

Letting $\beta_1 = 1.5 - u$ and $\gamma_1 = 0.5 + u$, the denominator of β_2 and γ_2 can be uniquely determined from (5.7). Identifying which values of $\{\kappa_0, \kappa_1, r\}$ correspond to zero denominators means solving

$$0 = \beta_1\gamma_1\kappa_0\kappa_1 - \sin^2(\theta),$$
$$0 = (\frac{3}{2} - u)(\frac{1}{2} + u)\kappa_0\kappa_1 - \sin^2\left(\lambda\kappa_0 + (1-\lambda)\kappa_1\right),$$

$$0 = (\frac{3}{2} - \frac{r+1}{r+2})(\frac{1}{2} + \frac{r+1}{r+2})\kappa_0\kappa_1 - \sin^2\left(\lambda\kappa_0 + (1-\lambda)\kappa_1\right),$$
$$0 = \frac{(r+4)(3r+4)\kappa_0\kappa_1}{4(r+2)^2} - \sin^2\left(\frac{r((1+r)\kappa_1 - \kappa_0) + (1+r)(\kappa_0 - \kappa_1)\ln(1+r)}{r^2}\right).$$

A closed form expression for the solutions to this equation, w.r.t. the parameters $\{\kappa_0, \kappa_1, r\}$, has not been identified. The complexity of the equation suggests no such form exists and solutions should instead be obtained via numerical procedures [33].

However, by using an alternative defining set of parameters $\{\theta, u, t\}$ the zero denominator GCS curves can be identified easily. Solving (5.7) to zero corresponds to:

$$\begin{aligned} 0 =& \beta_1\gamma_1\kappa_0\kappa_1 - \sin^2(\theta), \\ 0 =& \beta_1\gamma_1(\theta + (1-\lambda t))(\theta - \lambda t) - \sin^2(\theta), \\ 0 =& \beta_1\gamma_1\theta^2 - \sin^2(\theta) + \beta_1\gamma_1\theta(1-2\lambda)t - \beta_1\gamma_1(1-\lambda)\lambda t^2 \end{aligned} \tag{6.2}$$

where

$$\beta_1 = \tfrac{3}{2} - u$$
$$\gamma_1 = \tfrac{1}{2} + u$$
$$\lambda = \begin{cases} \frac{1}{2} & \text{if } u = \frac{1}{2}, \\ \frac{1-u}{2u-1}\left(\frac{u}{2u-1}\ln\left(\frac{u}{1-u}\right) - 1\right) & u \neq \frac{1}{2}. \end{cases}$$

Thus the variation of $D(\beta_1, \gamma_1)$ is quadratic in t.

6.2.1 Identifying divergent regions in a GCS class

By looking at the variation of $D(\beta_1, \gamma_1)$ within a GCS class, the zeros can be identified as the solutions to the quadratic equation (6.2):

$$t = t_0 \pm d = \frac{\theta(1-2\lambda)}{2(1-\lambda)\lambda} \pm \frac{\sqrt{\beta_1\gamma_1\theta^2 - 4(1-\lambda)\lambda\sin^2(\theta)}}{\sqrt{\beta_1\gamma_1}2(1-\lambda)\lambda}. \tag{6.3}$$

Considering the distribution of the second shape factors for a GCS class, divergent regions are identified to occur at the values $t = t_0 \pm d$ in (6.3). Taking Figure 6.4 as an example of a second shape factors' distribution within a class, away from divergent regions the distribution is approximately linear. Imposing this linear distribution around the divergent regions will provide stability. Thus a linear interpolation, symmetric about the two divergent points is created.

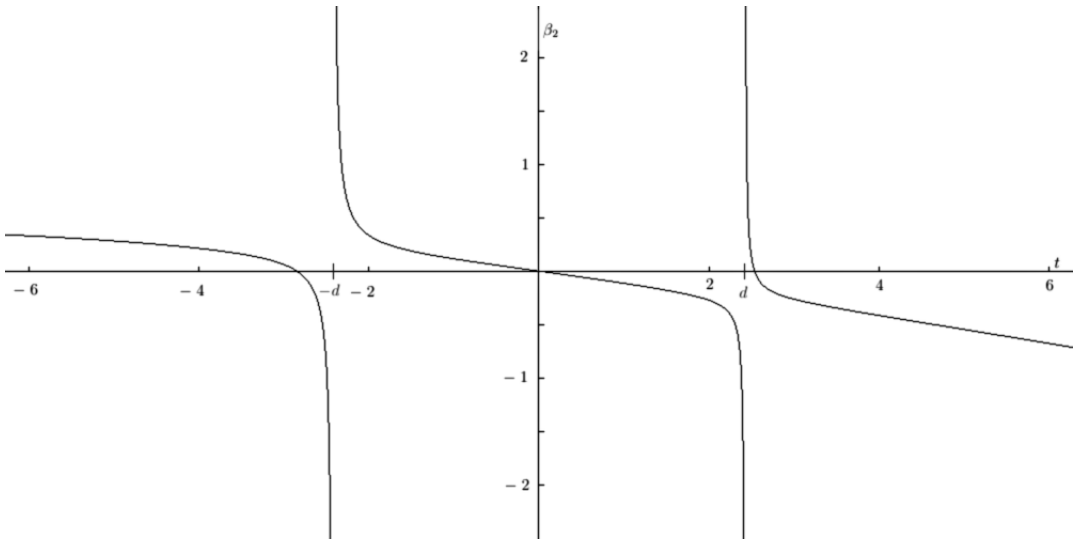

Figure 6.4: β_2 values for $\mathcal{C}_{0,\frac{\pi}{2}}(t)$. Divergence occurs local to $t = t_0 \pm d = 0 \pm 2\sqrt{(\frac{\pi}{2})^2 - 1}$.

6.2.2 Linear interpolating across divergent regions

Let the β_2 values at $t = t_0 - 2d, t_0, t_0 + 2d$, derived from (6.3), be denoted by b_{-1}, b_0, b_1 respectively. Then the linear interpolation is defined by the following equation (see Figure 6.5).

$$\beta_2(t) = \begin{cases} b_0 + \frac{t_0 - t}{2d}(b_{-1} - b_0) & \text{if } t \in (t_0 - 2d, t_0) \\ b_0 + \frac{t_0 - t}{2d}(b_1 - b_0) & \text{if } t \in (t_0, t_0 + 2d) \\ \beta_2(\frac{1}{2} + u, \frac{3}{2} - u) & \text{otherwise.} \end{cases} \tag{6.4}$$

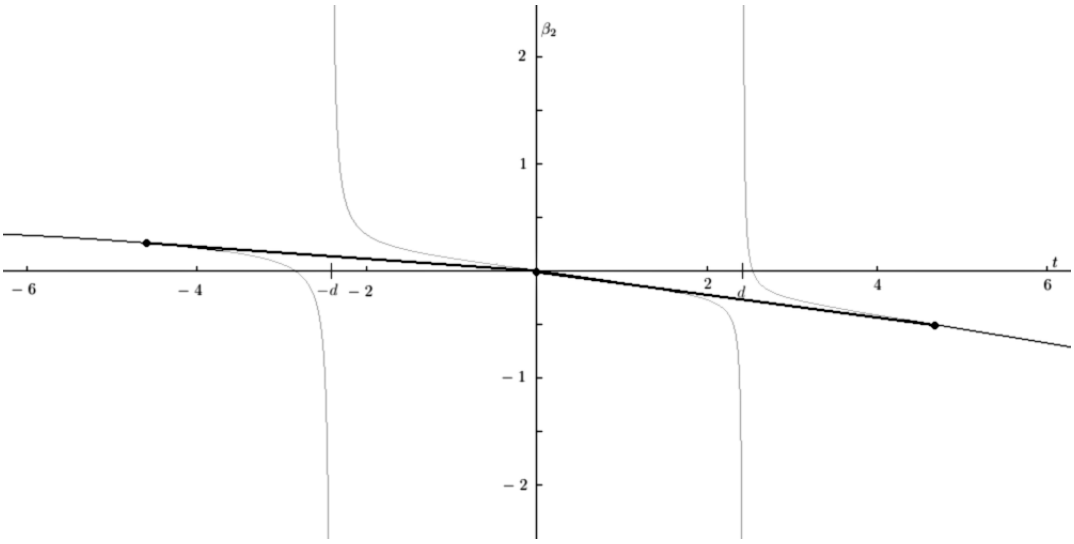

Figure 6.5: β_2 values for $\mathcal{C}_{0,\frac{\pi}{2}}(t)$ linearly interpolated across $t = t_0 - 2d, t_0, t_0 + 2d$.

Similarly, a linear interpolation of the γ_2 values in a GCS class is created. Let the γ_2 values at $t = t_0 - 2d, t_0, t_0 + 2d$ be denoted by g_{-1}, g_0, g_1 respectively. Then

$$\gamma_2(t) = \begin{cases} g_0 + \frac{t_0 - t}{2d}(g_{-1} - g_0) & \text{if } t \in (t_0 - 2d, t_0) \\ g_0 + \frac{t_0 - t}{2d}(g_1 - g_0) & \text{if } t \in (t_0, t_0 + 2d) \\ \gamma_2(\frac{1}{2} + u, \frac{3}{2} - u) & \text{otherwise.} \end{cases} \tag{6.5}$$

Defining the values of the second shape factors (β_2, γ_2) in this way prevents divergent behaviour and gives C^0 continuity for the distribution of these values whenever any of the parameters $\{\theta, u, t\}$ are varied.

As a consequence of not using the G^3 second shape factors, G^3 continuity of the approximation is no longer guaranteed. The method does still achieve G^2 continuity however, by construction. Whenever $|t - t_0| > 2d$ the G^3 shape factors are used and hence the approximation will be G^3 continuous. The approximation method is therefore referred to as "G^{2+}".

6.3 Assigning bounds

To ensure only high quality approximations are formed, bounding restrictions on the input parameters are often made [14] [23] [21] as previously noted in section 6.1.1. For example, the winding angle in practical road design is restricted to $|\theta| \leq \frac{\pi}{2}$ [21]. Similarly in [23], the winding angle is restricted to $|\theta| \leq \frac{\pi}{2}$ which was arrived at empirically. In [14], when creating a practical solution to span generation using GCS curves, the shape factor is restricted so that $|\lambda - \frac{1}{2}| < 0.4$.

6.3.1 Determining Bounds

In order to ensure the G^{2+} method yields only acceptable approximations, bounds on $\{\theta, u, t\}$ are imposed, as suggested by the insight gained from section 6.1.1. The winding angle is

restricted to $|\theta| \leq \frac{\pi}{2}$, the shape factor u is bounded by $|u - \frac{1}{2}| \leq 0.4$ and t limited to $|t| \leq \pi$. These bounds were arrived at using existing restrictions and empirical evidence. The winding angle restriction agrees with bounds from [21] and [23]. The bounds on u and t were arrived at as a result of observations motivated with the intention of maximising the region of admissible values.

If the GCS lies within these bounds, it is claimed that the G^{2+} method will always yield an acceptable approximation. Analysis related to this claim is dealt with in Chapter 7. If a GCS lies outside these bounds a process of subdividing the curve can lead to a collection of GCS segments that lie inside.

An approximation to the original GCS can then be formed by joining the subdivided curve approximations. Applying the appropriate translations, the subdivided GCSs can interpolate corresponding intermediary points of the original GCS. These subdivided curves can then be rotated to interpolate tangent directions. As each approximation is G^2 continuous (w.r.t. the GCS) this will form a G^2 continuous curve.

6.3.2 Convergence of subdivision inside the bounds

To confirm that subdivision is a valid technique for the G^{2+} method, it must be shown that any GCS can be subdivided to yield curves inside the bounds. The effect of subdivision with relation to the parameters $\{\theta, u, t\}$ is examined here to support the claim of convergence. Explicit details of a specific subdivision scheme which does converge can be found in Appendix 2.

Effect of subdivision on θ

Recall the winding angle is given by $\theta = \int_0^1 \kappa(s)ds$. A similar parameter, $\bar{\theta} = \int_0^1 |\kappa(s)|ds$, describes the *total winding angle* of the GCS where $\bar{\theta} \geq |\theta|$. Subdividing the curve at λ gives the values of $\bar{\theta}$ for the two subdivisions: $\bar{\theta}_1 = \int_0^\lambda |\kappa(s)|ds$, $\bar{\theta}_2 = \int_\lambda^1 |\kappa(s)|ds$.

Excluding the case of a straight line, when $\kappa(s) \equiv 0$ and thus $\theta = 0$, $\kappa(s)$ is a non-zero ($\kappa(s) \not\equiv 0$) monotonic function (see section 2.7.2). Since $0 < \lambda < 1$ and $\bar{\theta}_1, \bar{\theta}_2$ are integrals of positive functions it follows that $\bar{\theta}_1, \bar{\theta}_2 > 0$ and $\bar{\theta} = \bar{\theta}_1 + \bar{\theta}_2$. Thus $\bar{\theta}_1, \bar{\theta}_2 < \bar{\theta}$ and the total winding angle of a GCS, $\bar{\theta}$, always decreases in size after subdivisions. The bound of $|\theta| \leq \frac{\pi}{2}$ may be satisfied by ensuring that $\bar{\theta} \leq \frac{\pi}{2}$.

Effect of subdivision on u

Since u is related to r, via (3.3), an equivalent analysis on the effect of the shape factor r is considered. The effect of subdivision on r is outlined in Table 3.1 (see section 3.5). The case of a Cornu spiral, when $r = 0$, is excluded since $|u - \frac{1}{2}| = 0 < 0.4$.

The effect on the first subdivision is $r_1 = \lambda r$ and since $0 < \lambda < 1$ clearly $|r_1| < |r|$. For the second shape factor consider this sequence of inequalities:

$$r > -1$$

$$\lambda r > -\lambda$$

$$(1 + \lambda r) > (1 - \lambda)$$

$$1 > \frac{(1-\lambda)}{(1+\lambda r)}.$$

Since the shape factor of the second subdivision is $r_2 = r\frac{(1-\lambda)}{(1+\lambda r)}$ it has been shown that $|r_2| < |r|$. Thus the size of the shape factor, $|r|$, always decreases in size after subdivisions. By ensuring that $|r|$ is sufficiently small, the value of $|u - \frac{1}{2}|$ may also be bounded (since $r = 0$ implies $u = \frac{1}{2}$).

Effect of subdivision on t

Recall the parameter $t = (\kappa_0 - \kappa_1)$ represents the change in end curvatures. The effect of subdivision on t is again outlined in Table 3.1. The case of a circular segment, when $t = 0$ and thus $\kappa_0 = \kappa_1$, is excluded since $|t| = 0 < \pi$.

Let the curvature at λ be κ_λ. Then the values of t after subdivision are:

$$t_1 = (\kappa_0 - \kappa_\lambda)\lambda \qquad\qquad t_2 = (\kappa_\lambda - \kappa_1)(1-\lambda)$$

Since $\kappa(s)$ is a monotonic function (see section 2.7.2) and $0 < \lambda < 1$,

$$|t_1| = |(\kappa_0 - \kappa_\lambda)||\lambda| < |(\kappa_0 - \kappa_\lambda)| < |(\kappa_0 - \kappa_1)| = |t|,$$
$$|t_2| = |(\kappa_\lambda - \kappa_1)||1-\lambda| < |(\kappa_\lambda - \kappa_1)| < |(\kappa_0 - \kappa_1)| = |t|.$$

It follows that $|t_1|, |t_2| < |t|$ and thus $|t|$ always decreases in size after subdivisions.

6.3.3 A subdivision routine

The existence of a subdivision scheme, that divides a GCS into a number of segments such that each curve lies within identified bounds, is useful for guaranteeing a successful approximation with the G^{2+} method. An example of such a scheme is presented in Appendix 2. Given any GCS with $\{\theta, u, t\}$ outside the bounds, it is therefore possible to subdivide the curve until each segment lies within the bounds.

A concern with using a subdivision process may lie with the number of subdivisions required. Data proliferation can occur under certain circumstances. Scenarios include choosing the acceptable tolerance to an unachieveable precision and attempting to find approximations to unfeasable GCSs, which occur as $\mu \to 0$ (section 4.2.1), $\theta \to \pm\infty$, $u \to 0, 1$ or $t \to \pm\infty$. A consequence of these conditions is that the subdivision process will generate an excessive number of curve segments. To prevent this from happening the tolerance should be increased (the suggested value is 5% see section 4.2.1). Otherwise the feasability of the GCS should be questioned and a different GCS should be considered.

6.4 Algorithm summary of the G^{2+} method

The following steps summarise the G^{2+} approximation method:

1. Given a GCS, transform $\{\kappa_0, \kappa_1, r\} \rightarrow \{\theta, u, t\}$

using (3.4), (3.3) and (3.5).

2. Check the parameters satisfy $|\theta| \leq \frac{\pi}{2}, |t| \leq \pi$ and $|u - \frac{1}{2}| \leq 0.4$.

If they do not, apply a subdivision scheme until the bounds are met.

3. Define the first shape factors as $\beta_1 = \frac{3}{2} - u$, $\gamma_1 = \frac{1}{2} + u$ from (6.1).

4. Calculate the second shape factors (β_2, γ_2) from (6.4) and (6.5).

5. The approximation is given by the quintic B$\acute{e}$zier with control

points defined in (5.3).

6.5 Summary

This chapter presented a new method to approximate the GCS with a G^{2+} quintic Bézier. This G^{2+} method built upon the G^3 method of Chapter 5 and was improved by removing the need for a numerical search, reducing the computational cost of constructing an approximation. The GCS was defined in terms of a new set of parameters $\{\theta, u, t\}$ and bounds on these values were identified. If a GCS lies outside these bounds then a process of subdivision can yield a set of piecewise curves all of which lie inside the bounds. The next chapter aims to argue that, given any GCS curve within the specified bounds, the G^{2+} approximation method will be acceptable.

Chapter 7

Analysis of the G^{2+} Method

This chapter presents an analysis of the G^{2+} method by examining the claim that it always yields acceptable approximations. It begins by giving a formal mathematical condition of the claim. It is then argued that an analytical verification to this claim is unavailable and so instead a numerical approach is taken. Details of this numerical approach are presented and by using symmetrical properties of the GCS the complexity is reduced. The results of the analysis are then presented. This is followed by an explanation of what these results reveal about the effectiveness of the G^{2+} method.

7.1 Guaranteeing a satisfactory approximation

The effectiveness of an approximation is measured by the metric ϵ. However, this only provides a measure for a single GCS. In order to measure the effectiveness of an approximation method the metric should indicate its performance across a variety of GCS curves. This leads to the definition of a new metric which measures the accuracy of an approximation method across a range of GCS curves.

7.1.1 The error metric ($\bar{\epsilon}$)

The metric that measures the G^{2+} approximation accuracy for any GCS is the 3-dimensional function:

$$\bar{\epsilon} : \mathbb{R}^3 \rightarrow \mathbb{R}$$

$$\{\theta, t, u\} \mapsto \epsilon$$

which maps the GCS with input parameter $\{\theta, u, t\}$ to the error, ϵ. The domain of the function can be restricted to lie within the bounds outlined in Chapter 6. This is because a subdivision routine is applied to every GCS outside these bounds until the subdivided curves all lie inside the bounds (see section 6.3).

Thus the metric may be rewritten as:

$$\bar{\epsilon} : \mathbf{D} \rightarrow \mathbb{R}$$

$$\{\theta, t, u\} \mapsto \epsilon$$

where,

$$\mathbf{D} = \left\{(\theta, t, u) : -\frac{\pi}{2} \leq \theta \leq \frac{\pi}{2}, -\pi \leq t \leq \pi, 0.1 \leq u \leq 0.9\right\}.$$

Thus to show the G^{2+} method always yields acceptable approximations it is enough to verify that $\bar{\epsilon}(\mathbf{x}) \leq 0.05$ (see section 4.2.1) for all $\mathbf{x} \in \mathbf{D}$.

7.1.2 Symmetries

By exploiting symmetrical properties of the GCS the size of the domain, $\mathbf{D}$, can be reduced. This is done by finding equivalent approximations via isometries (section 2.2) of the GCS.

Two specific isometric transformations are examined in detail. By showing that the approximation to the transformed curve is equivalent to the transformation of the approximated curve an equivalence relationship is formed. Thus only one of these isometric curves needs to be calculated.

The two isometries that are utilised were discovered from the effect of reflection on the curvature profile. Consider the curvature profiles of two planar curves as $\kappa_1(s)$ for $s \in [0, S_1]$ and $\kappa_2(s)$ for $s \in [0, S_2]$ respectively. Isometries of the curve, relating to reflections, can be determined if $S_1 = S_2$ and $\kappa_1(s) = -\kappa_2(s)$ or $\kappa_1(s) = -\kappa(S_2 - s)$ for $s \in [0, S_1]$ (see Appendix C for proof).

The isometric transformations relating these curvature profiles are now examined in turn.

Isometry 1

Consider the transformation, α, which reflects the curve in the x-axis that maps

$$\alpha : \mathbf{V}(t) \to \overline{\mathbf{V}(t)}$$

$$\{x, y\} \mapsto \{x, -y\}$$

Then

$$\alpha(\mathbf{V}'(t)) \to \overline{\mathbf{V}'(t)}$$

$$\alpha(\mathbf{V}''(t)) \to \overline{\mathbf{V}''(t)}$$

Assume a curve has curvature profile $\kappa(s)$ then the curvature of the curve transformed by α is (2.7)

$$\begin{aligned}\alpha(\kappa(s)) &= \alpha\left(\frac{x'(t)y''(t) - y'(t)x''(t)}{(x'(t)^2 + y'(t)^2)^{\frac{3}{2}}}\right) \\ &= \frac{x'(t)(-y''(t)) - (-y'(t))x''(t)}{(x'(t)^2 + (-y'(t))^2)^{\frac{3}{2}}} \\ \alpha(\kappa(s)) &= -\kappa(s)\end{aligned}$$

thus the transformation is isometric (Appendix C).

Applying α to the GCS, the effect on the input parameters is

$$\{\kappa_0, \kappa_1, r\} \rightarrow \{-\kappa_0, -\kappa_1, r\}$$

which can be verified by substitution into (3.2). Furthermore the effect of the alternate input parameters is

$$\{\theta, u, t\} \rightarrow \{-\theta, u, -t\}$$

which can again be verified by substitution into (3.3-5).

The G^{2+} approximation of the transformed curve thus takes the form (5.3)

$$\begin{aligned}
\alpha(\mathbf{V}_0) &= \overline{\mathbf{F}(0)} &&= \overline{\mathbf{V}_0} \\
\alpha(\mathbf{V}_1) &= \overline{\mathbf{F}(0)} + \frac{\beta_1}{5}\overline{\mathbf{F}'(0)} &&= \overline{\mathbf{V}_1} \\
\alpha(\mathbf{V}_2) &= \overline{\mathbf{F}(0)} + \big(\frac{\beta_2}{20} + \frac{2\beta_1}{5}\big)\overline{\mathbf{F}'(0)} + \frac{\beta_1^2}{20}\overline{\mathbf{F}''(0)} &&= \overline{\mathbf{V}_2} \\
\alpha(\mathbf{V}_3) &= \overline{\mathbf{F}(1)} + \big(\frac{\gamma_2}{20} - \frac{2\gamma_1}{5}\big)\overline{\mathbf{F}'(1)} + \frac{\gamma_1^2}{20}\overline{\mathbf{F}''(1)} &&= \overline{\mathbf{V}_3} \\
\alpha(\mathbf{V}_4) &= \overline{\mathbf{F}(1)} + \frac{-\gamma_1}{5}\overline{\mathbf{F}'(1)} &&= \overline{\mathbf{V}_4} \\
\alpha(\mathbf{V}_5) &= \overline{\mathbf{F}(1)} &&= \overline{\mathbf{V}_5}.
\end{aligned}$$

This is equivalent to applying the isometric transformation to the approximation because

$$\begin{aligned}\alpha(\mathbf{V}(t)) &= \alpha(\sum_{i=0}^{5} B_i^5(t)\mathbf{V}_i) \\ &= \sum_{i=0}^{5} B_i^5(t)\overline{\mathbf{V}_i}\end{aligned}$$

Thus the approximation to the transformed curve is equivalent to the transformation of the approximated curve. The isometry is summarised by the effect on the control points:

$$\alpha(\{\mathbf{V}_0, \mathbf{V}_1, \mathbf{V}_2, \mathbf{V}_3, \mathbf{V}_4, \mathbf{V}_5\}) \rightarrow \{\overline{\mathbf{V}_0}, \overline{\mathbf{V}_1}, \overline{\mathbf{V}_2}, \overline{\mathbf{V}_3}, \overline{\mathbf{V}_4}, \overline{\mathbf{V}_5}\}.$$

This verifies the equivalence for GCS approximations with

$$\{\theta, u, t\} \leftrightarrow \{-\theta, u, -t\}.$$

Isometry 2

Consider the transformation, α, which describes the curve backwards. That is, for $\mathbf{V}(t)$ with $t \in [0, 1]$ let

$$\alpha : \mathbf{V}(t) \rightarrow \mathbf{V}(1 - t)$$

Then

$$\alpha(\mathbf{V}'(t)) \to -\mathbf{V}'(1-t)$$

$$\alpha(\mathbf{V}''(t)) \to \mathbf{V}''(1-t)$$

Assume a curve has curvature profile $\kappa(s)$ then the curvature of the curve transformed by α is (2.7)

$$\begin{aligned}\alpha(\kappa(s)) &= \alpha\left(\frac{x'(t)y''(t) - y'(t)x''(t)}{(x'(t)^2 + y'(t)^2)^{\frac{3}{2}}}\right) \\ &= \frac{(-x'(1-t))y''(1-t) - (-y'(1-t))x''(1-t)}{((-x'(1-t))^2 + (-y'(1-t))^2)^{\frac{3}{2}}} \\ \alpha(\kappa(s)) &= -\kappa(1-s)\end{aligned}$$

thus the transformation is isometric (Appendix C).

Applying α to the GCS, the effect on the input parameters is

$$\{\kappa_0, \kappa_1, r\} \to \{-\kappa_1, -\kappa_0, \frac{-r}{1+r}\}$$

which can be verified by substitution into (3.2). Furthermore the effect of the alternate input parameters is

$$\{\theta, u, t\} \to \{-\theta, 1-u, t\}$$

which can again be verified by substitution into (3.3-5).

The isometric transformation to the approximation thus takes the form

$$\begin{aligned}\alpha(\mathbf{V}(t)) &= \alpha(\sum_{i=0}^{5} B_i^5(t)\mathbf{V}_i) \\ &= \alpha(\sum_{i=0}^{5} \frac{5!}{(5-i)!i!}(1-t)^{5-i}t^i\mathbf{V}_i) \\ &= \sum_{i=0}^{5} \frac{5!}{(5-i)!i!}(1-(1-t))^{5-i}(1-t)^i\mathbf{V}_i \\ &= \sum_{i=0}^{5} \frac{5!}{(5-i)!i!}t^{5-i}(1-t)^i\mathbf{V}_i\end{aligned}$$

By re-labelling the indices of the summation, $j = 5 - i$, then

$$\begin{aligned}\alpha(\mathbf{V}(t)) &= \sum_{j=0}^{5} \frac{5!}{j!(5-j)!}(1-t)^{5-j}t^j\mathbf{V}_{5-j} \\ &= \sum_{j=0}^{5} B_j^5(t)\mathbf{V}_{5-j}.\end{aligned}$$

This shows the effect of the transformation on the control points, that is

$$\alpha(\mathbf{V}_i) = \mathbf{V}_{5-i}, \quad i = 0 \ldots 5.$$

This can be shown to be equivalent to the G^{2+} approximation of the transformed curve which is given by (5.3)

$$
\begin{aligned}
\alpha(\mathbf{V}_0) &= \mathbf{F}(1) && = \mathbf{V}_5 \\
\alpha(\mathbf{V}_1) &= \mathbf{F}(1) - \frac{\alpha(\beta_1)}{5}\mathbf{F}'(1) && (= \mathbf{V}_4) \\
\alpha(\mathbf{V}_2) &= \mathbf{F}(1) - \big(\frac{\alpha(\beta_2)}{20} + \frac{2\alpha(\beta_1)}{5}\big)\mathbf{F}'(1) + \frac{\alpha(\beta_1)^2}{20}\mathbf{F}''(1) && (= \mathbf{V}_3) \\
\alpha(\mathbf{V}_3) &= \mathbf{F}(0) - \big(\frac{\alpha(\gamma_2)}{20} - \frac{2\alpha(\gamma_1)}{5}\big)\mathbf{F}'(0) + \frac{\alpha(\gamma_1)^2}{20}\mathbf{F}''(0) && (= \mathbf{V}_2) \\
\alpha(\mathbf{V}_4) &= \mathbf{F}(0) - \frac{-\alpha(\gamma_1)}{5}\overline{\mathbf{F}'(0)} && (= \mathbf{V}_1) \\
\alpha(\mathbf{V}_5) &= \mathbf{F}(0) && = \mathbf{V}_0.
\end{aligned}
$$

A symmetry between the shape factors $(\beta_1, \gamma_1; \beta_2, \gamma_2)$ is observed. In order to equate the control points of the approximation, conditions on the shape factors must be satisfied. That is

$$\alpha(\beta_1) = \gamma_1 \qquad \text{and} \qquad \alpha(\gamma_1) = \beta_1,$$

which in turn will yield

$$\alpha(\beta_2) = -\gamma_2 \qquad \text{and} \qquad \alpha(\gamma_2) = -\beta_2,$$

by construction. For the G^{2+} approximation this condition can be validated since

$$
\begin{aligned}
\alpha(\beta_1) &= \alpha(1.5 - u) & \qquad \alpha(\gamma_1) &= \alpha(0.5 + u) \\
&= 1.5 - (1 - u) & &= 0.5 + (1 - u) \\
&= 0.5 + u & &= 1.5 - u \\
&= \gamma_1 & &= \beta_1.
\end{aligned}
$$

Thus the approximation to the transformed curve is equivalent to the transformation of the approximated curve. The isometry is summarised by the effect on the control points:

$$\alpha(\{\mathbf{V}_0, \mathbf{V}_1, \mathbf{V}_2, \mathbf{V}_3, \mathbf{V}_4, \mathbf{V}_5\}) \rightarrow \{\mathbf{V}_5, \mathbf{V}_4, \mathbf{V}_3, \mathbf{V}_2, \mathbf{V}_1, \mathbf{V}_0\}.$$

This verifies the equivalence for GCS approximations with

$$\{\theta, u, t\} \leftrightarrow \{-\theta, 1-u, t\}.$$

By demonstrating isometries of approximations it has been shown that isometric GCSs yield the same error. This result can be described using the notation from section 7.1.1 as:

$$\bar{\epsilon}(\{\theta, u, t\}) = \bar{\epsilon}(\{-\theta, u, -t\}) = \bar{\epsilon}(\{-\theta, 1-u, t\})$$

Thus only one of the errors for these isometric curves needs to be calculated. If t is negative then the equivalent GCS with input parameters $\{-\theta, u, -t\}$ can be used. Similarly, if θ is negative then the equivalent GCS with input parameters $\{-\theta, 1-u, t\}$ can be used. Thus it is reasonable to only consider the curves where $\theta, t \geq 0$. The size of $\mathbf{D}$ can thus be reduced to

$$\mathbf{D} = \left\{(\theta, t, u) : 0 \leq \theta \leq \frac{\pi}{2}, 0 \leq t \leq \pi, 0.1 \leq u \leq 0.9\right\}.$$

7.1.3 Forming sequential approximations ($\bar{\epsilon_n}$) to ($\bar{\epsilon}$)

To show that the G^{2+} approximation algorithm always yields satisfactory approximations it must be shown that $\bar{\epsilon}(\mathbf{x}) \leq 0.05$ for all $\mathbf{x} \in \mathbf{D}$. This is equivalent to showing that the maximum value of $\bar{\epsilon}$, that is the upper bound, is less than 0.05. The methods available to determine the upper bound of a function depend on the properties of the function.

To begin, observe that $\bar{\epsilon}$ is a C^0 continuous function as it is the composition of C^0 functions (see section 4.2 and section 6.2.2). Furthermore, the function $\bar{\epsilon}$ is not C^1 continuous (see section 4.2) and thus partial derivatives are intractable. Analytic methods to calculate bounds are thus almost impossible.

Instead sequential simplified approximations $\bar{\epsilon_n}$ to $\bar{\epsilon}$ are considered. If these simplified approximations converge uniformly to $\bar{\epsilon}$ then in the limit the two functions, and thus the bounds of the functions, will coincide. The bounds of $\bar{\epsilon}$ can thus be realised as the bound of $\bar{\epsilon_n}$ as $n \to \infty$.

It is known that any C^0 function may be approximated by piecewise linear segments and the approximation will uniformly converge as the number of segments increases [36]. For a 3-dimensional C^0 function an analogous piecewise tri-linear segment approximation can be applied and will also converge uniformly as the number of tri-linear segments increase.

The approximations $\bar{\epsilon}_n$ of $\bar{\epsilon}$ are considered as its piecewise tri-linear interpolation. Each sequential approximation considers half the distance for each one parameter linear segment and thus is defined on $(2^n + 1)^3$ lattice points (see Figure 7.1(a-b)).

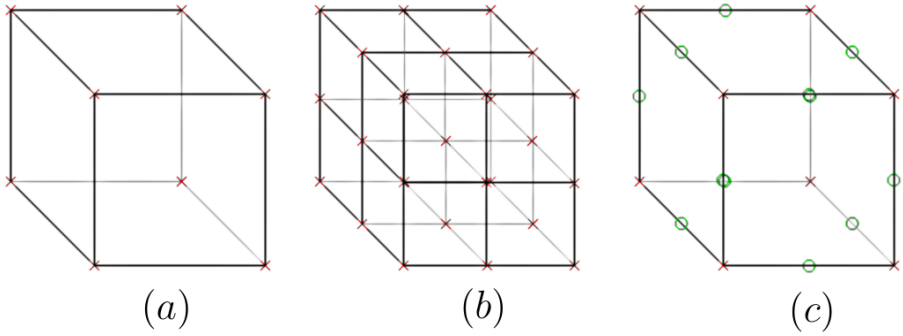

Figure 7.1: [a-b] Visualisation of the piecewise tri-linear interpolation (a) n=0, (b) n=1. [c] The cirles are considered for calculating the error of the tri-linear approximation.

7.1.4 Calculating a bound for $\bar{\epsilon}$

Bounds for the function $\bar{\epsilon}_n$ can be easily calculated. Let the upper bound be denoted χ_n. This value is found from the maximum value of all lattice points, i.e. if $\mathbf{L}_n$ is the set of all lattice points then

$$\chi_n = \max_{\mathbf{x} \in \mathbf{L}_n} \{\bar{\epsilon}(\mathbf{x})\}.$$

For each sequential approximation an error value ν_n, which represents an error of $\bar{\epsilon}_n$ to $\bar{\epsilon}$, can be generated. A comparison between the values of $\bar{\epsilon}$ and $\bar{\epsilon}_n$ at the midpoint of all neighbouring lattice points is considered (see Figure 7.1(c)). The error, ν_n, is then given as the largest difference in values between all those pairs of points. Thus if $\mathbf{N}_n$ is the set of all neighbouring lattice points and $(\mathbf{x}, \mathbf{y}) \in \mathbf{N}_n$ are neighbouring points (with $\mathbf{x}, \mathbf{y} \in \mathbf{L}_n$) then

$$\nu_n = \max_{(\mathbf{x},\mathbf{y}) \in \mathbf{N}_n} \left\{ \bar{\epsilon}_n \left(\frac{\mathbf{x}+\mathbf{y}}{2} \right) - \bar{\epsilon} \left(\frac{\mathbf{x}+\mathbf{y}}{2} \right) \right\}$$

$$= \max_{(\mathbf{x},\mathbf{y})\in\mathbf{N}_n} \left\{ \frac{\bar{\epsilon}(\mathbf{x}) + \bar{\epsilon}(\mathbf{y})}{2} - \bar{\epsilon}\left(\frac{\mathbf{x}+\mathbf{y}}{2}\right) \right\}.$$

From these values a bound for $\bar{\epsilon}$ can be estimated as $\xi_n = \chi_n + \nu_n$. In the limit as $n \to \infty$, $\nu_n \to 0$ and ξ_n tends towards the true bound of $\bar{\epsilon}$.

7.1.5 Table of bounds

The following table presents the bounding data for $\bar{\epsilon_n}$ when $n = 0, \ldots, 7$:

n	$\lvert L_n \rvert$	χ_n	ν_n	ξ_n
0	2^3	0.025721	0.011702	0.037423
1	3^3	0.025721	0.009192	0.034913
2	5^3	0.025721	0.006524	0.032245
3	9^3	0.025721	0.003389	0.029110
4	17^3	0.025721	0.002040	0.027761
5	33^3	0.025721	0.001118	0.026839
6	65^3	0.025721	0.000546	0.026267
7	129^3	0.025721	0.000528	0.026249

Table 7.1: Table of bounds for sequential approximations of $\bar{\epsilon_n}$ to $\bar{\epsilon}$.

From Table 7.1, the values of ξ_n appear to be tending to a value less than 0.05. Thus this data provides strong evidence that $\bar{\epsilon}(\mathbf{x}) \leq 0.05 \ \forall \mathbf{x} \in \mathbf{D}$ which would indicate that the proposed method does produce acceptable approximations for every GCS within the bounds.

Furthermore, the values of ν_n are getting smaller as expected. This is because the sequential approximations of $\bar{\epsilon}_n$ are converging to $\bar{\epsilon}$. Most interestingly, the values for χ_n remain constant. This is because the maximum value in the lattice was found at the boundary point $\bar{\epsilon}(\frac{\pi}{2}, 0.1, \pi)$

for all $n = 0, \ldots, 7$. This suggests that the upper bound is achieved at this point.

7.2 Concluding remarks from analysis

This chapter began by giving a formal mathematical condition, involving $\bar{\epsilon}$, for the proof of the claim that a successful approximation of a GCS using the G^{2+} method can be guaranteed within bounded paramters. Properties of $\bar{\epsilon}$ meant that a suitable way to determine its bounds was to consider sequential approximations to itself. The bounds could then be inferred from the limit of these approximations. To reduce the size of the domain of $\bar{\epsilon}$, isometric properties of the GCS were utilised.

The data obtained from these approximations is presented in Table 7.1. The results give strong evidence that the G^{2+} does indeed always provide a satisfactory approximation for bounded parameters. Furthermore, the upper limit of $\bar{\epsilon}$, i.e. the GCS with the highest error, appears to lie on the boundary at $\{\theta, u, t\} = \{(\frac{\pi}{2}, 0.1, \pi)\}$ and has an error of $\epsilon = 0.026$.

Chapter 8

Examples

Examples of the G^3 and G^{2+} approximation methods developed in Chapters 5 and 6 are examined in this chapter. A discussion of the approximation performance as well as comparisons between these methods and the existing C^2 Hermite approximation is presented. This C^2 Hermite interpolant is equivalent to the second order s-power series of [22]. Reasoning behind this choice of method for comparison was discussed in the literature review (see section 1.3).

To begin, approximations to the quarter circle are examined. Then approximations of the GCS from Figure 5.2, where the initial G^3 approximation was unsatisfactory, are developed. This GCS requires a G^3 search and detailed steps of this search are given. Following this a graph which shows the accuracy for the initial $G^3(1,1)$ approximation across a range of Cornu spirals is presented. This indicates how often a search may be required. Then approximations of the GCS identified as the upper bound of $\bar{\epsilon}$ (see Chapter 7) are compared. Finally, the last figure displays a comparison of each approximation methods accuracy across a variety of GCS curves.

The chapter finishes with a discussion on observations from the examples. Of particular interest is the insight gained when comparing each methods effectiveness and hence suitability to serve as a replacement for a GCS.

8.1 The quarter circle

An approximation to the quarter circle can be found by considering the normalised GCS with $\kappa_0 = \kappa_1 = \frac{\pi}{2}, r = 0$. The initial $G^3(1,1)$ approximation assigns the parameters $\beta_1 = \gamma_1 = 1$. This gives $\beta_2 = 0.0078$, $\gamma_2 = -0.0078$ (see section 5.2.2) and yields an error of $\epsilon = 5.5 \cdot 10^{-4}$. For this GCS the G^{2+} approximation gives exactly the same Bézier curve and thus also yields an error of $\epsilon = 5.5 \cdot 10^{-4}$. Finally, a C^2 Hermite approximation, where $\beta_1 = \gamma_1 = 1$ and $\beta_2 = \gamma_2 = 0$, gives an error of $\epsilon = 0.0030$. An illustration of these approximations and their curvature profiles is given in Figure 8.1. Note that all are acceptable approximations, as $\epsilon < 5\%$, but the C^2 Hermite interpolant is not quite as accurate as the other two.

8.2 A G^3 search

Next, the normalised GCS with $\kappa_0 = 0.36, \kappa_1 = 2.7, r = 0.1$ from Figure 5.2 is reconsidered. The initial $G^3(1,1)$ approximation gives β_2 and γ_2 as -5.82 and -2.00 respectively. This produces an error $\epsilon = 0.23$ and hence the approximation is unsatisfactory. Out of interest, the denominator (5.6) can be calculated as $D(\beta_1, \gamma_1) = -0.028$ suggesting that this initial approximation may be unsuitable (see section 5.2.6).

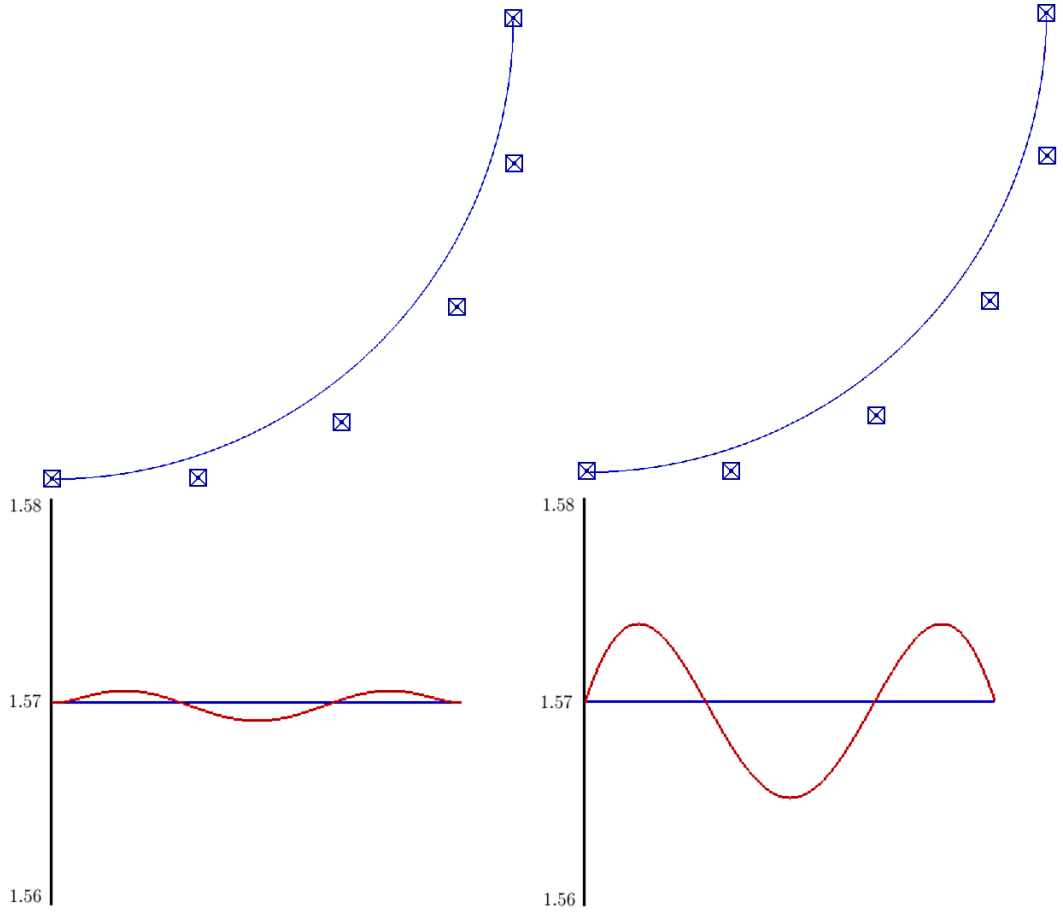

Figure 8.1: Approximations to the quarter-circle with their curvature profiles. [Left] The G^3 and G^{2+} approximation. [Right] The C^2 Hermite approximation.

Thus a search on $\epsilon(\beta_1, \gamma_1)$ should be implemented. An initial value for the search must be decided. Solving $\beta_2 = \gamma_2 = 0$ using Maple © software 14 solutions were found [37] all of which give orthogonal first and second derivatives. Six solutions were non-real, three were negative, and three had a denominator less than 0.1. Recalling β_1, γ_1 must be real, positive and give a denominator greater than 0.1 to lie in $\mathcal{P}$, these solutions were discarded. Of the remaining 2 solutions the one with least error had a starting value of (β_1, γ_1)=$(0.73, 1.11)$ resulting in (β_2, γ_2)=$(-0.22, -0.11)$. This gave an error of $\epsilon = 0.074$.

A search was then implemented from this initial point. After a single iteration of Powell's

method a better approximation with (β_1, γ_1) = $(0.70, 1.15)$ and (β_2, γ_2) = $(1.33, 0.51)$ resulted in an error of $\epsilon = 0.0089$. An illustration of the searching domain is given in Figure 8.2.

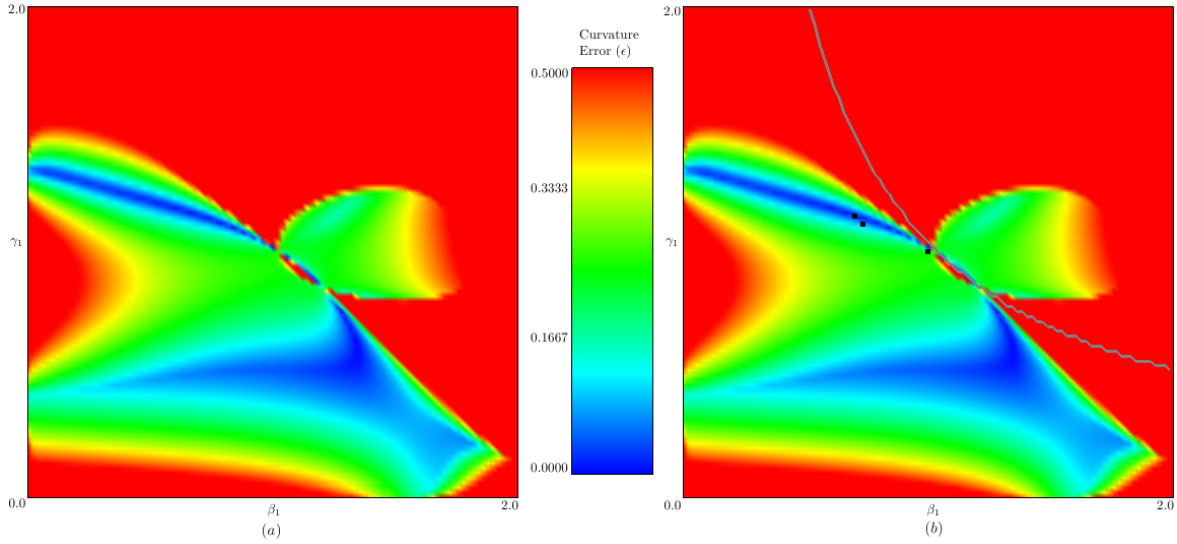

Figure 8.2: [a] The error (ϵ) as a function of the shape parameters β_1, γ_1 define the search domain. [b] This image highlights the region with a zero denominator (grey) and the sequential approximations of $[\beta_1, \gamma_1] = [1, 1], [0.73, 1.11]$ and $[0.70, 1.15]$ (black dots).

The G^{2+} method gives $\beta_1 = 0.98, \gamma_1 = 1.02$ and $\beta_2 = 0.10, \gamma_2 = 0.31$ with an error of $\epsilon = 0.0087$. The C^2 Hermite approximation yields an error of $\epsilon = 0.025$. These approximations are given in Figure 8.3.

8.3 The initial G^3 approximation

The next figure, Figure 8.4, illustrates the effectiveness of the $G^3(1, 1)$ initial approximation for a range of Cornu spirals $(r = 0)$ and hence gives an insight into how often a search is required. The values for initial and end curvatures, κ_0 and κ_1, are varied from $-\pi$ to π with a resolution of 100×100 points.

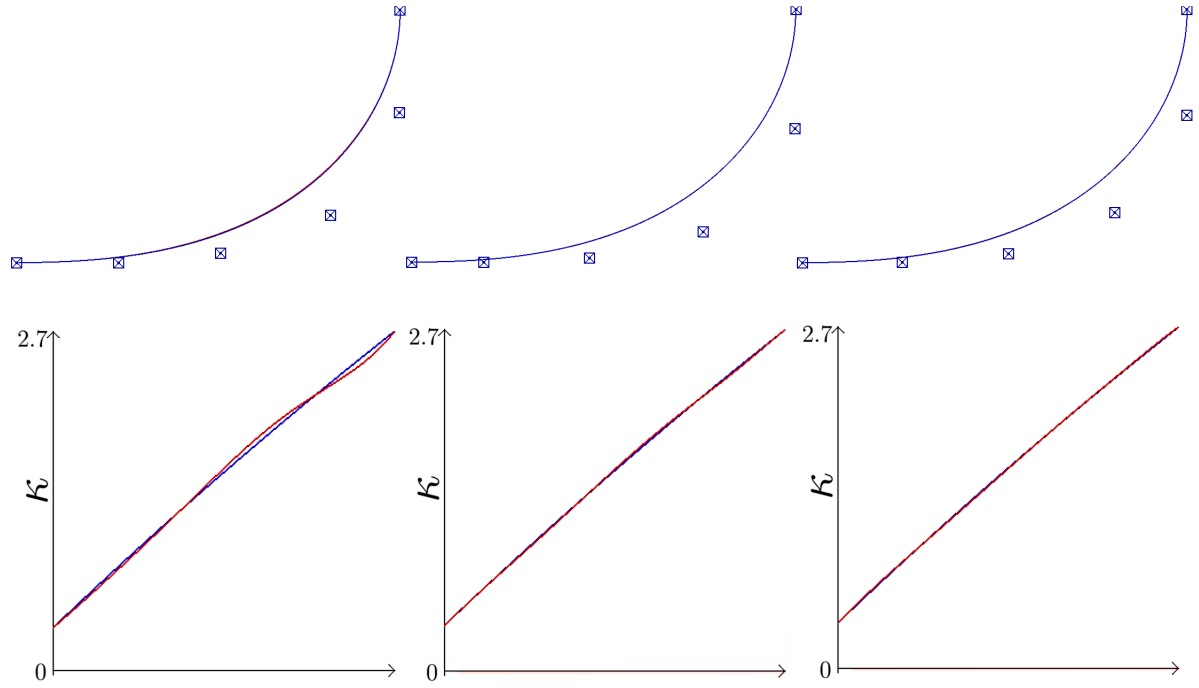

Figure 8.3: [Top] Approximations (blue) to the normalised GCS (red) ($\kappa_0 = 0.36, \kappa_1 = 2.7, r = 0.1$) [Bottom] Corresponding curvature profiles. [left - Hermite, middle - $G^3(0.70, 1.15)$, right - G^{2+}].

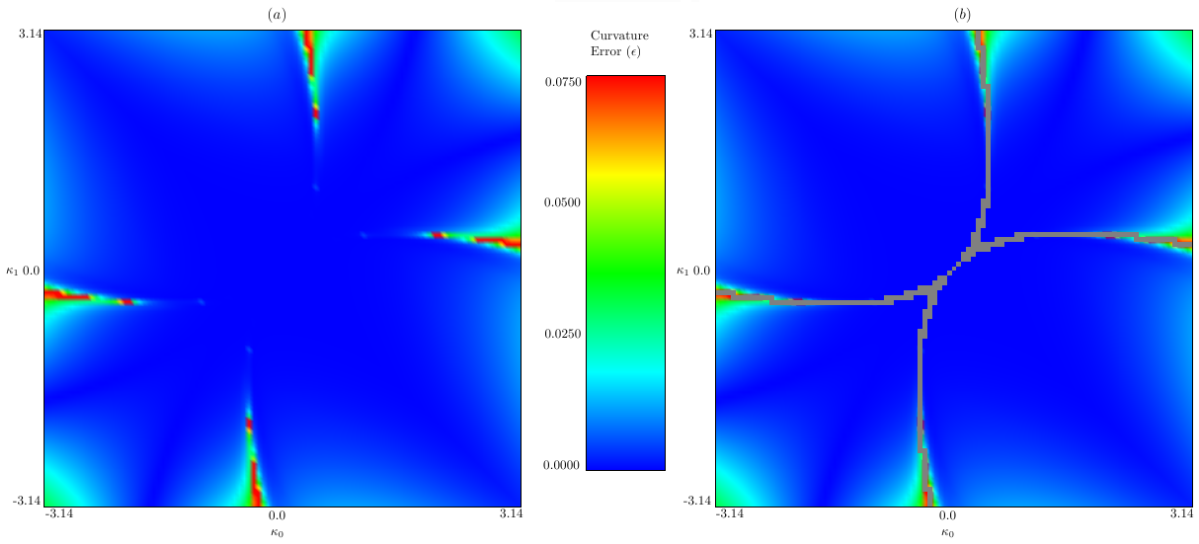

Figure 8.4: [a] The error (ϵ) of the initial $\beta_1 = \gamma_1 = 1$ G^3 approximation for a variety of Cornu spirals with $\kappa_0, \kappa_1 = -\pi \ldots \pi$. [b] This image highlights when the denominator function is zero (grey).

From the figure it is clear that most approximations result in an immediate satisfactory approximation (99% of the region). In fact the only places where the error exceeds the tolereance,

$\epsilon >= 0.05$, are local to a zero denominator. This is shown by the proximity to the grey region where the denominator equals zero.

8.4 The GCS identified as the upper bound to $\bar{\epsilon}$

This example considers the GCS corresponding to the upper bound of $\bar{\epsilon}$ from section 7.1.5, that is when $\{(\theta, t, u) = (\frac{\pi}{2}, \pi, 0.1)\}$ which has values of $\{(\kappa_0, \kappa_1, r) = (2.15, -1.00, -0.889)\}$. The $G^3(1,1)$ approximation gives $\beta_2 = -1.34, \gamma_2 = -2.96$. The G^{2+} method gives $\beta_1 = 1.4, \gamma_1 = 0.6$ and $\beta_2 = -1.33, \gamma_2 = -1.12$. The error values for ϵ are $3\%, 29\%$ and 31%, for the G^{2+}, G^3 and Hermite models respectively. An illustration of these approximations and their curvature profiles is given in Figures 8.5 and 8.6.

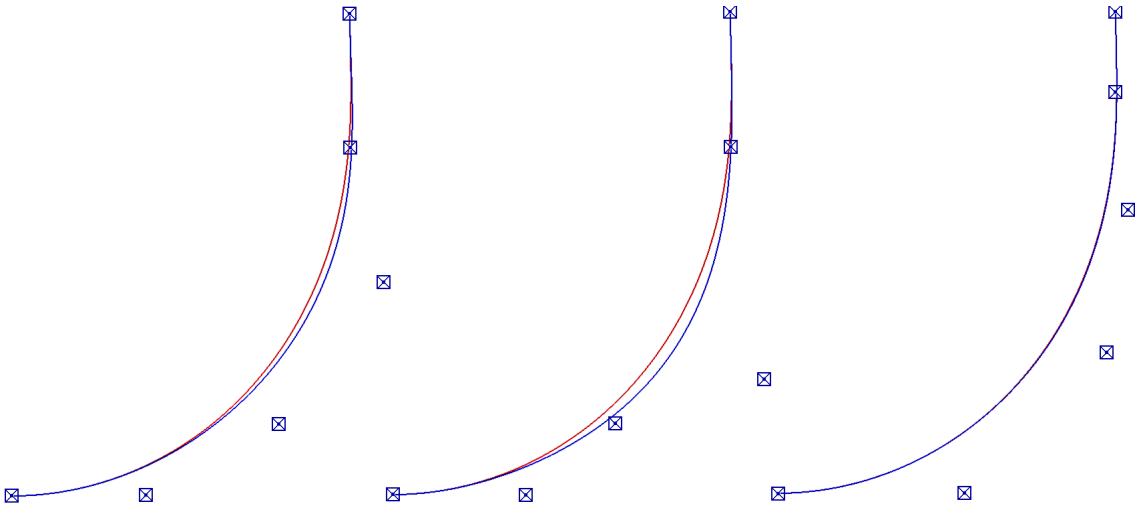

Figure 8.5: Approximations (blue) to the GCS (red) $\{(\kappa_0, \kappa_1, r) = (2.15, -1.00, -0.889)\}$ [left - Hermite, middle - $G^3(1,1)$, right - G^{2+}].

It is clear that the G^{2+} approximation outperforms both of the other methods, when approximating this GCS, since it is the only method which yields an acceptable approximation.

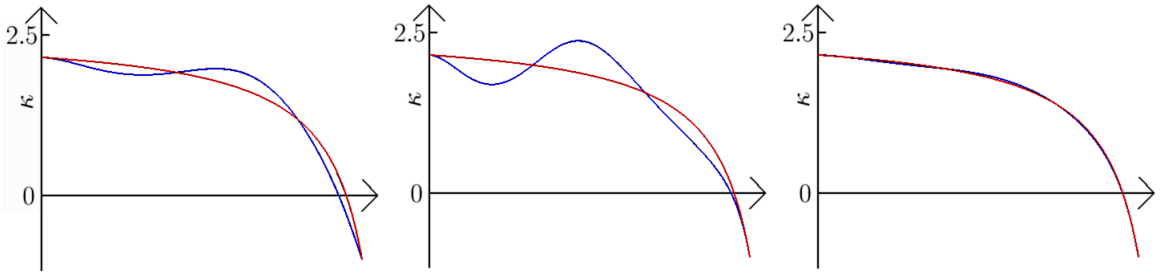

Figure 8.6: Curvature profiles of the approximations (blue) to the GCS (red) $\{(\kappa_0, \kappa_1, r) = (2.15, -1.00, -0.889)\}$ [left - Hermite, middle - $G^3(1,1)$, right - G^{2+}].

8.5 Comparison of each methods accuracy

Figure 8.7 demonstrates how the approximation methods perform across a variety of GCSs. For each plot the variation in the x-axis corresponds to a change in shape factor $0.1 < u < 0.9$. The variation in the y-axis corresponds to a change in the parameter $0 < t < \pi$. The winding angle is fixed to coincide with the bounding plane $\theta = \frac{\pi}{2}$ and the hot-cold scaled errors range from blue ($\epsilon = 0.00$) to red ($\epsilon = 0.05$). The images were created with a resolution of 80×100 points (80 points in u ,100 points in t).

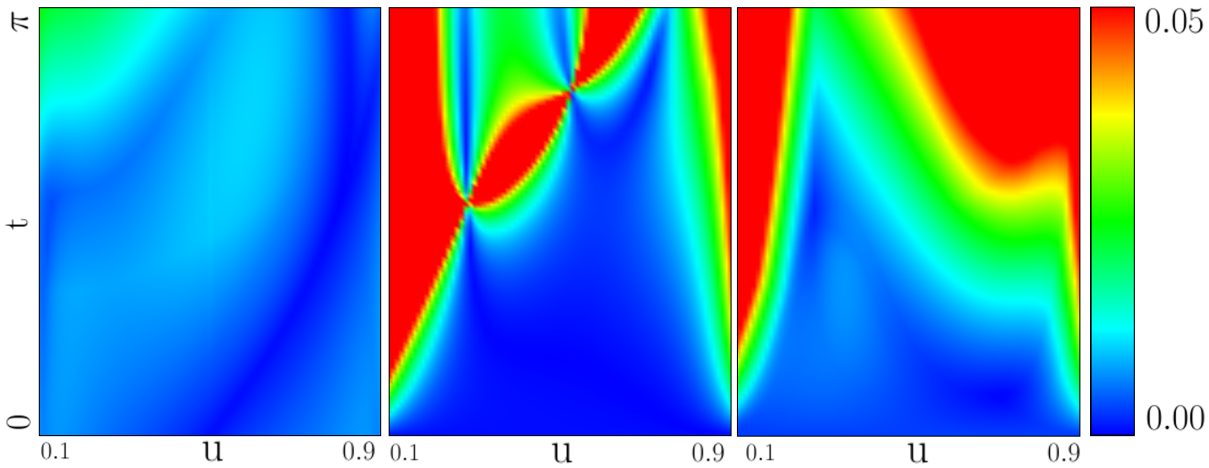

Figure 8.7: The error ϵ in the approximation methods [left - G^{2+}, middle - $G^3(1,1)$, right - Hermite] when $\theta = \frac{\pi}{2}$.

It is clear from the figure that the G^{2+} approximation outperforms both of the other methods. The graph corresponding to G^{2+} does not contain any regions of unsatisfactory approximations whereas both the $G^3(1,1)$ and Hermite methods fail as t approaches π and as u approaches the boundaries $u = 0.1$ and $u = 0.9$.

The inflexibility of the Hermite method does not take into account the rising complexity of the shape of the GCS as t and u approach the boundary edges. The $G^3(1,1)$ approximation appears to provide a better approximation except for the isolated regions in the upper center and upper right. These regions of unsatisfactory approximations correspond to denominator values approaching zero (see Figure 6.1 for correlation). Furthermore, the method also fails as u approaches 0.1 and 0.9. This is a consequence of not varying the shape factors β_1, γ_1, as suggested by the G^3 optimal search (see section 6.1.2).

8.6 Concluding remarks

This chapter examined the perfomance of three different methods to approximate the GCS. The method which gave the most accurate approximations (sections 8.1,8.2 and 8.4) was the G^{2+} method. The $G^3(1,1)$ approximation generally outperformed the Hermite interpolant with exception to the GCS in section 8.2. However, by applying a search routine a better approximation than the Hermite interpolant was found. Figure 8.4 from section 8.3 supported the claim that small denominator (5.6) values cause poor $G^3(1,1)$ approximations. Away from these divergent regions, the approximation was acceptable.

Section 8.5 provided more insight into how each approximation performed across a variety of GCSs. The G^{2+} method outperformed both the other methods since it did not contain any unacceptable approximations. Figure 8.7 highlighted a weakness of the $G^3(1,1)$ and Hermite methods. They assign static values of $\beta_1 = \gamma = 1$ for the shape factors contrary to the G^{2+} method. The divergence issues of the $G^3(1,1)$ approximation can also be seen in the figure.

This chapter has shown, by studying various examples, that the G^{2+} method outperfoms the other two. Overall it provided more accurate approximations. However, when compared to using a G^3 optimal search (section 6.1) approximations are generally less accurate. This is not a concern since a G^3 search suffers from stability issues and thus can not be considered robust. Furthermore, the G^{2+} method does not require a computationally expensive search routine. Therefore the G^{2+} approximation is the most suitable method to approximate a GCS. It has been demonstrated to be robust and efficient and therefore worthy of consideration for CAD implementation.

Chapter 9

Conclusions

The goal of this research was to develop an efficient and robust polynomial approximation to the GCS. The approximation procedure formulated in Chapter 6, the G^{2+} method, has demonstrated both of these properties and should therefore be considered for practical application. A summary of the research developed in the thesis is given next. This is followed with a condensed outline of the approximation procedure. The chapter finishes with some remarks on possible future research.

9.1 Research overview

The first stage of the research involved developing a foundation of knowledge on the relevant background material, so as to recast the initial research objectives inside a mathematically rigor-

ous framework. The material, found in Chapter 2, motivated the choice of polynomial approximation. It was argued to be a quintic polynomial in the Bézier form. Geometric constraints, namely G^2 end point interpolation, was also argued for. This left four degrees of freedom. Thus the goal of the research was realised as finding suitable values for these four degrees of freedom.

The next chapter examined the GCS in more detail. Various properties of the GCS were discussed which would be used later on in the thesis. These included normalisation, reparameterisation and subdivision. By applying an equivalence relation to the set of GCS curves, with respect to proportional scaling transformations, normalisation served to reduce the seven defining parameters of the GCS to three. The representation of these three parameters was then altered to reflect geometric features of the curve. These parameters would be used later in the construction of a G^{2+} approximation. Finally, GCS subdivision was introduced as means to improve accuracy by taking piecewise approximations to smaller segments. This would eventually be used to demonstrate the robustness of the G^{2+} method.

Chapter 4 detailed the error metric (ϵ) used to measure the accuracy of an approximation. It was decided that the error should compare the curvature profiles of the GCS and the Bézier approximation as discussed in [23]. This error was defined as the minimum of the absolute curvature difference and the absolute relative curvature difference. It was then possible to classify approximations as acceptable if this error was within some tolerance, i.e. $\epsilon < \mu$. This tolerance was chosen to be $\mu = 5\%$.

An initial procedure to produce approximations to the GCS, the G^3 method, was given in Chapter 5. To begin, the construction of a G^2 approximation with a quintic Bézier curve was derived. A further constraint, equating end curvature profile derivatives, led to the G^3 construction. Two

degrees of freedom, (β_1, γ_1) which correspond to the size of the end tangent vectors, remained. To conform with the representation of the GCS, the values of these parameters were chosen to equal 1. This formed an initial approximation. However, these values for the two free parameters were not always suitable; a consequence of a divergence in the construction procedure. To overcome this problem alternative values were found by performing a numerical search.

Issues associated with the use of a numerical search motivated the construction of the G^{2+} approximation. By avoiding the use of a numerical search, the efficiency of the approximation procedure could be established. This was also necessary for demonstrating the robustness of the algorithm.

Chapter 6 began by gathering insight into the free parameters (β_1, γ_1) from the results of a G^3 optimal search. These values were then modelled with a linear distribution with respect to the parameter u. To avoid the divergent behaviour of the second shape factors, (β_2, γ_2), a linear distribution (with respect to the parameter t) about the divergent regions was implemented. The chapter finished with a discussion of how GCS subdivision could be implemented to satisfy bounds on input parameters $\{\theta, u, t\}$. This completed the definition of the G^{2+} approximation procedure.

Chapter 7 presented an analysis of the G^{2+} approximation by examining the claim that it always yields acceptable approximations. To begin, a formal mathematical condition for this claim was derived. Analytic verification of this claim was not possible using standard derivative analysis. Instead a means of verifying this claim via a sequential numerical method was presented. The results of the analysis gave strong evidence that the G^{2+} method does indeed always provide an acceptable approximation.

Finally, Chapter 8 compared examples of three different approximation routines to a variety of GCS curves. The three methods examined were the C^2 quintic Hermite approximation, the G^3 method and the G^{2+} method. Approximations to the quarter circle and the GCS corresponding to the upper bound of $\bar{\epsilon}$ were given as well as an implementation of a G^3 search. Finally, a graph which illustrated the performance of each method across a range of GCS curves was presented. The examples presented indicate that the G^{2+} method outperforms the other two methods.

9.2 Outline of the G^{2+} approximation procedure

Figure 9.1 presents the approximation procedure in the format of a flow diagram.

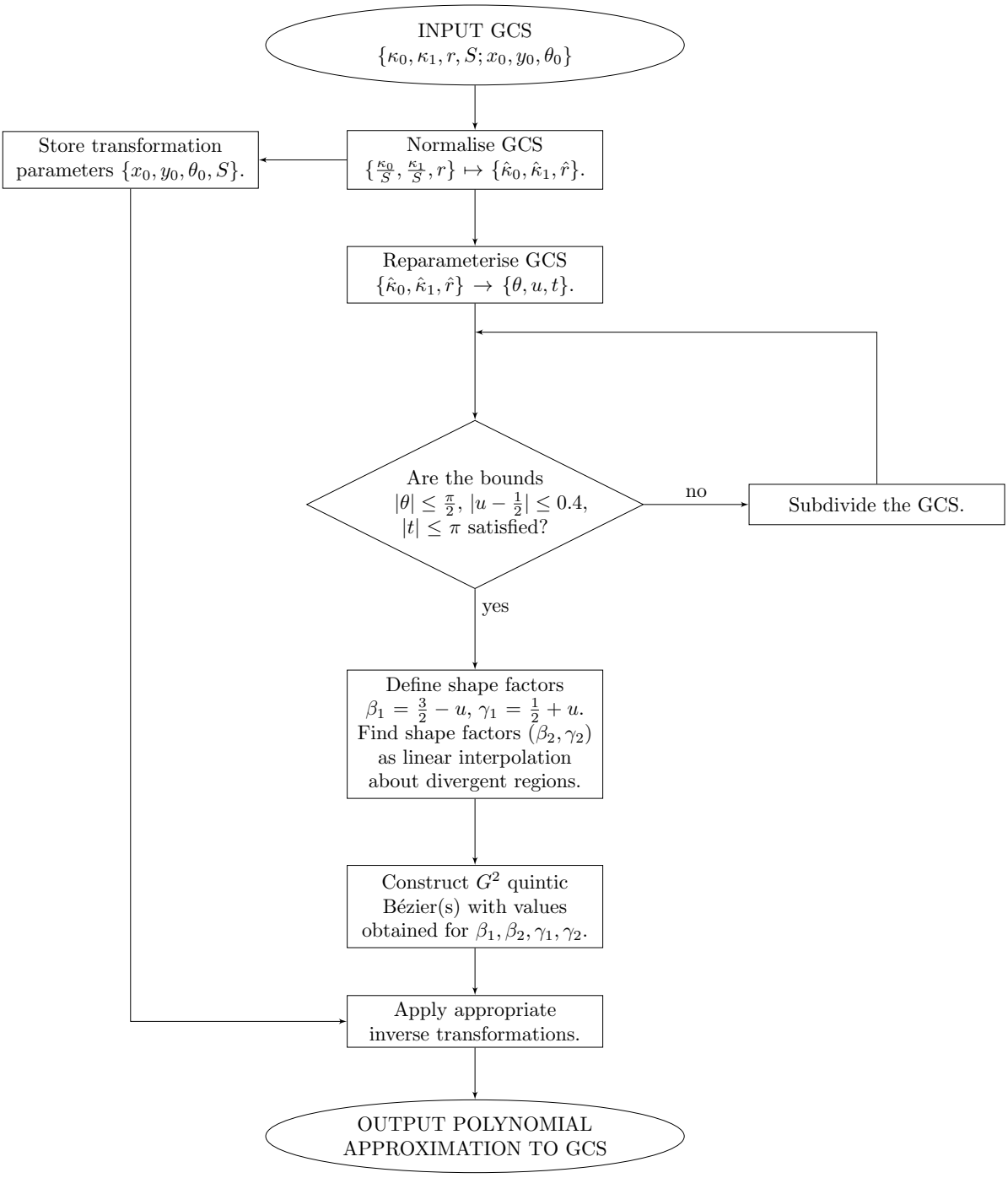

Figure 9.1: Flow diagram detailing the G^{2+} approximation procedure.

9.3 Concluding remarks

The research objectives of the thesis were to develop a procedure to replace the GCS with a suitable polynomial approximation. To justify its practicality for CAD implementation the procedure should demonstrate efficiency and robustness.

The G^{2+} approximation method has satisfied these objectives. The efficiency of the method is a consequence of removing the numerical search required to construct an approximation. By utilising a subdivision scheme, verification of the robustness was achieved. Thus given any GCS it is now possible to construct a polynomial approximation efficiently.

9.4 Future research

Rational representations of curves, such as rational Bézier and NURBS, were not considered in this thesis. Approximations to the GCS using rational curves, for example a rational cubic Bézier, could be a suitable area for future research. A lot of the techniques developed in this thesis may be easily adapted to accommodate a rational approximation.

Recall the initial motivation for creating a polynomial approximation to the GCS, from the introduction to the thesis. The desire was to able to create high quality curves for use in a CAD-CAM environment. Given geometric data, such as position and tangent directions, GCS curves may be found to interpolate these features [23]. However, the process of finding a suitable GCS is a relatively expensive process. An alternative approach is to apply a GCS-like approxima-

tion without having to find the actual GCS to interpolate the data. The consequence being an efficient system for interpolating data points with inherently fair curves.

In a practical setting, most curves are not planar (2D) but space (3D) curves. Since the GCS is a planar curve, an extension to space curves would be of interest with respect to high quality 3D curve constructions. Such a 3D curve has been made by finding the intersection of the projection of GCS curves in perpendicular planes [1]. However, extensive analysis into the properties of this curve construction has not been studied. Alternative construction techniques could also be formed that take into account a curves' torsion as well as its curvature. Polynomial approximations to these high quality 3D-GCS curves will then provide another interesting area of research.

Finally, another important area of research considers high quality surface constructions. It has been hypothesised that generating a surface with high quality curves and a suitable construction procedure can result in a high quality surface [1] [2]. Surfaces built with the GCS, or 3D-GCSs, may result in a high quality construction. It is hypothesised that by applying polynomial approximations to these GCS generated surfaces a high quality polynomial patch can be constructed efficiently. This could eventually lead to an efficient algorithm for constructing high quality surfaces with polynomial patches.

References

[1] Cripps, R.J. *Quality Surface Construction.* Journal of Prime Research in Mathematics Vol. 3, (2007),

[2] Cripps, R.J. and Lockyer, P.S. *Circle approximation for CADCAM using orthogonal C2 cubic B-splines.* The International Journal of Machine Tools and Manufacture 45, (2005), p1222-1229.

[3] Ball, A.A. *CAD: master or servant of engineering?*. The Mathematics of Surfaces VII, Information Geometers, (1997), p17-23.

[4] Ma, X. Cripps, R.J. *Shape preserving data reduction for 3D surface points*. Computer Aided Design, Vol. 43, Issue 8, (Aug 2011), p902-909.

[5] Cripps, R.J. *Algorithms to support point-based CADCAM*. International Journal of Machine Tools and Manufacture, Vol. 43, Iss. 4, (March 2003), p425432.

[6] Nielson, G.M. *Rectangular v-Splines.* Computer Graphics and Applications, IEEE Vol. 6, Issue 2, (Feb 1986), p35-40.

[7] Rando, T. and Roulier, J.A. *Measures of Fairness for Curves and Surfaces*. Designing fair curves and surfaces, Society for Industrial and Applied Mathematics, Philadelphia, (1994), p75-122.

[8] Farin, G. *Curves and Surfaces for Computer Aided Graphical Design*. Academic Press (2002).

[9] Farin, G. Rein, G. Sapidis, N.S. and Worsey, A.J. *Fairing Cubic B-Splines*. Computer Aided Geometric Design, Vol 4, (July 1986), p91-103.

[10] Walton, D.J. Meek, D.S. G^2 *curve design with a pair of Pythagorean Hodograph quintic spiral segments.* Computer Aided Geometric Design 24, (2007), p267-285.

[11] Walton, D.J. and Meek, D.S. *A planar cubic Bézier spiral*. Journal of Computational and Applied Mathematics 72, (1996), p85-100.

[12] Kurnosenko, A. *Two-point G^2 Hermite interpolation with spirals by inversion of hyperbola*. Computer Aided Geometric Design, Vol. 27, Issue 6, (August 2010), p474-481.

[13] Meek, D.S. and Walton, D.J. *Clothoid Spline Transition Spirals*. Mathematics of Computation, Vol 59, (July 1992), p117-133.

[14] Ali, J.M. Tookey, R.M. Ball, J.V. Ball, A.A. *The Generalised Cornu Spiral and its applications to span generation*. Journal of Computational and Applied Mathematics, (1994), p37-47.

[15] Guggenheimer, H.W. *Differential Geometry*. Dover Publications (1997).

[16] Davis, P.J. *Interpolation and Approximation.* Dover Publications, (1975).

[17] Goldapp, M. *Approximation of circular arcs by cubic polynomials*. Computer Aided Geometric Design 8, (1991), p227-238.

[18] Fang, L. *Circular arc approximation by quintic polynomial curves*. Computer Aided Geometric Design 15, (1998), p843-861.

[19] Baumgarten, C. and Farin, G. *Approximation of Logarithmic Spirals*. Computer Aided Geometric Design, (1997), p515-532.

[20] Heald, M.A. *Rational Approximations for the Fresnel Integrals*. Mathematics of Computation, Vol. 44, No. 170, (April 1985), p459-461.

[21] Wang, L.Z. Miura, K.T. Nakamae, E. Yamamoto, T. Wang, T.J. *An approximation approach of the clothoid curve defined in the interval* $[0, \pi/2]$ *and its offset by free-form curves*. Computer Aided Design 33, (2001), p1049-1058.

[22] Sánchez-Reyes, J. and Chacón, J.M. *Polynomial approximation to clothoids via s-power series*. Computer Aided Design 35, (2003), p1305-1313.

[23] Cripps, R.J. Hussain, M.Z. and Zhu, S. *Smooth polynomial approximation of spiral arcs*. Journal of Computational and Applied Mathematics 233, (2010), p2227-2234.

[24] Zhu, S. *G^2-Approximation of Generalised Cornu Spirals by planar Bézier curves*. (Thesis), The University of Birmingham, (Dec 2004).

[25] Zwillinger, D. *Standard Mathematical Tables and Formulae.* Chapman and Hall/CRC Press LLC, (2003).

[26] Kühnel W. *Differential Geometry Curves-Surfaces-Manifolds*. American Mathematical Society (2006).

[27] Farouki, R.T. Sakkalis, T *Real rational curves are not 'unit speed'* Computer Aided Geometrical Design 8, (1991), p151-157.

[28] Barsky, B.A. and DeRose, T.D. *Geometric Continuity of Parametric Curves: Three Equivalent Characterizations*. IEEE Computer Graphics & Applications, (November 1989), p60-68.

[29] Craik, Alex D.D. *Prehistory of Faà di Bruno's Formula*. The American Mathematical Monthly, Vol. 112, No. 2, (Feb 2005), p119-130.

[30] Lopez, M.A. and Reisner, S. *Hausdorff approximation of convex polygons.* Computational Geometry 32, (2005), p139158.

[31] Cripps, R.J. Ball, A.A. *Orthogonal C^2 Cubic Spline Curves* Geometrical Modeling and Computing: Seattle 2003, Nashboro Press (2004).

[32] Cross, B. and Cripps, R.J. *G^3 quintic polynomial approximation for Generalised Cornu Spiral segments*. Journal of Computational and Applied Mathematics 236, (2012), p3111-3122.

[33] Press, W.H. Teukolsky, S.A. Vetterling W.T. Flannery B.P. *Numerical recipes in C, The art of scientific computing, Second edition* Cambridge University Press, (1994).

[34] Brieskorn, E. Knörrer, H. *Plane Algebraic Curves.* Birkäuser Boston (1981).

[35] Cross, B. and Cripps, R.J. *The Development of Polynomial Approximations to Generalised Cornu Spiral Segments*. 14th IMA Conference on Mathematics of Surfaces.

[36] Cheney, E.W. *Approximation Theory*. Chelsea Publishing Company, Second Edition (1982).

[37] *Maple 12,* Copyright 1991-2008, Waterloo Maple Inc.

Appendix A

The G^3 quintic Bézier construction

This appendix details the construction of the G^3 quintic Bézier by finding the solution to the G^3 constraints (5.5) at the end points as given in section 5.2.2. The equations to be solved are:

$$60\mathbf{F}(0) + (\beta_3 + 9\beta_2 + 36\beta_1)\mathbf{F}'(0) + (9\beta_1^2 + 3\beta_1\beta_2)\mathbf{F}''(0) + \beta_1^3\mathbf{F}'''(0) =$$
$$60\mathbf{F}(1) + (3\gamma_2 - 24\gamma_1)\mathbf{F}'(1) + 3\gamma_1^2\mathbf{F}''(1),$$

$$60\mathbf{F}(0) + (3\beta_2 + 24\beta_1)\mathbf{F}'(0) + 3\beta_1^2\mathbf{F}''(0) =$$
$$60\mathbf{F}(1) + (-\gamma_3 + 9\gamma_2 - 36\gamma_1)\mathbf{F}'(1) + (9\gamma_1^2 - 3\gamma_1\gamma_2)\mathbf{F}''(1) - \gamma_1^3\mathbf{F}'''(1).$$

Substituting the derivative data from section 5.1.1 gives:

$$60\begin{pmatrix}0\\0\end{pmatrix}+(\beta_3+9\beta_2+36\beta_1)\begin{pmatrix}1\\0\end{pmatrix}+(9\beta_1^2+3\beta_1\beta_2)\begin{pmatrix}0\\\kappa_0\end{pmatrix}+\beta_1^3\begin{pmatrix}-\kappa_0^2\\(\kappa_1-\kappa_0)(1+r)\end{pmatrix}=$$
$$60\begin{pmatrix}x\\y\end{pmatrix}+(3\gamma_2-24\gamma_1)\begin{pmatrix}\cos\theta\\\sin\theta\end{pmatrix}+3\gamma_1^2\kappa_1\begin{pmatrix}-\sin\theta\\\cos\theta\end{pmatrix},$$

$$60\begin{pmatrix}0\\0\end{pmatrix}+(3\beta_2+24\beta_1)\begin{pmatrix}1\\0\end{pmatrix}+3\beta_1^2\begin{pmatrix}0\\\kappa_0\end{pmatrix}=60\begin{pmatrix}x\\y\end{pmatrix}+(-\gamma_3+9\gamma_2-36\gamma_1)\begin{pmatrix}\cos\theta\\\sin\theta\end{pmatrix}$$
$$+(9\gamma_1^2-3\gamma_1\gamma_2)\kappa_1\begin{pmatrix}-\sin\theta\\\cos\theta\end{pmatrix}-\gamma_1^3\left[\begin{pmatrix}-\sin\theta\\\cos\theta\end{pmatrix}\frac{(\kappa_1-\kappa_0)}{(1+r)}-\begin{pmatrix}\cos\theta\\\sin\theta\end{pmatrix}\kappa_1^2\right].$$

Rearranging in terms of $\beta_2, \beta_3, \gamma_2, \gamma_3$ for each component, the four equations are:

$$9\beta_2+\beta_3-3\cos\theta\gamma_2=$$
$$-36\beta_1+\beta_1^3+\kappa_0^2+60x-24\gamma_1\cos\theta+3\gamma_1^2(-\sin\theta\kappa_1),$$

$$3\beta_1\kappa_0\beta_2-3\sin\theta\gamma_2=$$
$$-9\beta_1^2\kappa_0-\beta_1^3(\kappa_1-\kappa_0)(1+r)+60y-24\gamma_1\sin\theta+3\gamma_1^2\cos\theta\kappa_1,$$

$$3\beta_2+\cos\theta\gamma_3-9\cos\theta\gamma_2+3\gamma_1(-\sin\theta\kappa_1)\gamma_2=$$
$$-24\beta_1+60x-36\gamma_1\cos\theta+9\gamma_1^2(-\sin\theta\kappa_1)+\gamma_1^3\sin\theta\frac{(\kappa_1-\kappa_0)}{(1+r)}+\gamma_1^3\cos\theta\kappa_1^2,$$

$$\sin\theta\gamma_3-9\sin\theta\gamma_2+3\gamma_1\cos\theta\kappa_1\gamma_2=$$
$$-3\beta_1^2\kappa_0+60y-36\gamma_1\sin\theta+9\gamma_1\cos\theta\kappa_1-\gamma_1^3\cos\theta\frac{(\kappa_1-\kappa_0)}{(1+r)}+\gamma_1^3\sin\theta\kappa_1^2.$$

In matrix form this is equivalent to:

$$\begin{bmatrix} 9 & 1 & -3\cos\theta & 0 \\ 3\beta_1\kappa_0 & 0 & -3\sin\theta & 0 \\ 3 & 0 & -3(\gamma_1\sin\theta\kappa_1 + 3\cos\theta) & \cos\theta \\ 0 & 0 & 3(\gamma_1\cos\theta\kappa_1 + 3\sin\theta) & \sin\theta \end{bmatrix} \begin{bmatrix} \beta_2 \\ \beta_3 \\ \gamma_2 \\ \gamma_3 \end{bmatrix} = \begin{bmatrix} \delta_1 \\ \delta_2 \\ \delta_3 \\ \delta_4 \end{bmatrix}.$$

where,

$$\begin{aligned}
\delta_1 &= -36\beta_1 + \beta_1^3\kappa_0^2 + 60x - 24\gamma_1\cos\theta + 3\gamma_1^2(-\sin\theta\kappa_1), \\
\delta_2 &= -9\beta_1^2\kappa_0 - \beta_1^3(\kappa_1 - \kappa_0)(1+r) + 60y - 24\gamma_1\sin\theta + 3\gamma_1^2\cos\theta\kappa_1, \\
\delta_3 &= -24\beta_1 + 60x - 36\gamma_1\cos\theta + 9\gamma_1^2(-\sin\theta\kappa_1) + \gamma_1^3\sin\theta\frac{(\kappa_1 - \kappa_0)}{(1+r)} + \gamma_1^3\cos\theta\kappa_1^2, \\
\delta_4 &= -3\beta_1^2\kappa_0 + 60y - 36\gamma_1\sin\theta + 9\gamma_1\cos\theta\kappa_1 - \gamma_1^3\cos\theta\frac{(\kappa_1 - \kappa_0)}{(1+r)} + \gamma_1^3\sin\theta\kappa_1^2.
\end{aligned}$$

Thus the system of equations can be solved for $\beta_2, \beta_3, \gamma_2, \gamma_3$ since:

$$\begin{bmatrix} \beta_2 \\ \beta_3 \\ \gamma_2 \\ \gamma_3 \end{bmatrix} = \begin{bmatrix} 9 & 1 & -3\cos\theta & 0 \\ 3\beta_1\kappa_0 & 0 & -3\sin\theta & 0 \\ 3 & 0 & -3(\gamma_1\sin\theta\kappa_1 + 3\cos\theta) & \cos\theta \\ 0 & 0 & 3(\gamma_1\cos\theta\kappa_1 - 3\sin\theta) & \sin\theta \end{bmatrix}^{-1} \begin{bmatrix} \delta_1 \\ \delta_2 \\ \delta_3 \\ \delta_4 \end{bmatrix}. \qquad \text{(A1.1)}$$

Labeling the matrix to be inverted as $\mathbf{A}$ where

$$\mathbf{A} = \begin{bmatrix} 9 & 1 & -3\cos\theta & 0 \\ 3\beta_1\kappa_0 & 0 & -3\sin\theta & 0 \\ 3 & 0 & -3(\gamma_1\sin\theta\kappa_1 + 3\cos\theta) & \cos\theta \\ 0 & 0 & 3(\gamma_1\cos\theta\kappa_1 - 3\sin\theta) & \sin\theta \end{bmatrix},$$

then

$$\mathbf{A}^{-1} = \frac{1}{|\mathbf{A}|}\mathbf{A}^{\dagger}$$

where $|\mathbf{A}|$ and $\mathbf{A}^{\dagger}$ denote the determinant and the adjoint of matrix A respectively. Note that the inverse only exists when $|\mathbf{A}| \neq 0$. This zero determinant will correspond to the zero denominator of (5.6).

The determinant can be calculated as:

$$\begin{aligned} |A| =& -1 \left| \begin{bmatrix} 3\beta_1\kappa_0 & -3\sin\theta & 0 \\ 3 & -3(\gamma_1\sin\theta\kappa_1 + 3\cos\theta) & \cos\theta \\ 0 & 3(\gamma_1\cos\theta\kappa_1 - 3\sin\theta) & \sin\theta \end{bmatrix} \right| \\ =& -3\beta_1\kappa_0 \left| \begin{bmatrix} -3(\gamma_1\sin\theta\kappa_1 + 3\cos\theta) & \cos\theta \\ 3(\gamma_1\cos\theta\kappa_1 - 3\sin\theta) & \sin\theta \end{bmatrix} \right| \\ & -3\sin\theta \left| \begin{bmatrix} 3 & \cos\theta \\ 0 & \sin\theta \end{bmatrix} \right| \\ =& -3\beta_1\kappa_0(-3\gamma_1\kappa_1\sin^2\theta - 9\cos\theta\sin\theta - 3\gamma_1\kappa_1\cos^2\theta + 9\cos\theta\sin\theta) - 9\sin^2\theta \end{aligned}$$

$$=9(\beta_1\gamma_1\kappa_0\kappa_1 - \sin^2\theta).$$

Denoting the adjoint matrix as:

$$\mathbf{A}^\dagger = \begin{bmatrix} a_0^0 & a_0^1 & a_0^2 & a_0^3 \\ a_1^0 & a_1^1 & a_1^2 & a_1^3 \\ a_2^0 & a_2^1 & a_2^2 & a_2^3 \\ a_3^0 & a_3^1 & a_3^2 & a_3^3 \end{bmatrix},$$

then,

$$\begin{aligned}
a_0^0 &= 0, & a_2^0 &= 0, \\
a_0^1 &= 3\gamma_1\kappa_1, & a_2^1 &= 3\sin\theta, \\
a_0^2 &= -3\sin^2\theta, & a_2^2 &= -3\beta_1\kappa_0\sin\theta, \\
a_0^3 &= 3\sin\theta\cos\theta, & a_2^3 &= 3\beta_1\kappa_0\cos\theta, \\
a_1^0 &= 9[\beta_1\gamma_1\kappa_0\kappa_1 - \sin^2\theta], & a_3^0 &= 0, \\
a_1^1 &= 9[\sin\theta\cos\theta - 3\gamma_1\kappa_1], & a_3^1 &= 9[3\sin\theta - \gamma_1\kappa_1\cos\theta], \\
a_1^2 &= 9\sin\theta[3\sin\theta - \beta_1\kappa_1\cos\theta], & a_3^2 &= -3\beta_1\kappa_0[9\sin\theta - 3\gamma_1\kappa_1\cos\theta], \\
a_1^3 &= 9\cos\theta[\beta_1\kappa_0\cos\theta - 3\sin\theta], & a_3^3 &= 9[\beta_1\kappa_0(3\cos\theta + \gamma_1\kappa_1\sin\theta) - \sin\theta].
\end{aligned}$$

The inverse matrix, $\mathbf{A}^{-1}$, can thus be written as:

$$\frac{1}{3}\begin{bmatrix} 0 & \frac{\gamma_1\kappa_1}{[\beta_1\gamma_1\kappa_0\kappa_1-\sin^2\theta]} & \frac{-\sin^2\theta}{[\beta_1\gamma_1\kappa_0\kappa_1-\sin^2\theta]} & \frac{\sin\theta\cos\theta}{[\beta_1\gamma_1\kappa_0\kappa_1-\sin^2\theta]} \\ 3 & \frac{3[\sin\theta\cos\theta-3\gamma_1\kappa_1]}{[\beta_1\gamma_1\kappa_0\kappa_1-\sin^2\theta]} & \frac{3\sin\theta[3\sin\theta-\beta_1\kappa_1\cos\theta]}{[\beta_1\gamma_1\kappa_0\kappa_1-\sin^2\theta]} & \frac{3\cos\theta[\beta_1\kappa_0\cos\theta-3\sin\theta]}{[\beta_1\gamma_1\kappa_0\kappa_1-\sin^2\theta]} \\ 0 & \frac{\sin\theta}{[\beta_1\gamma_1\kappa_0\kappa_1-\sin^2\theta]} & \frac{-\beta_1\kappa_0\sin\theta}{[\beta_1\gamma_1\kappa_0\kappa_1-\sin^2\theta]} & \frac{\beta_1\kappa_0\cos\theta}{[\beta_1\gamma_1\kappa_0\kappa_1-\sin^2\theta]} \\ 0 & \frac{3[3\sin\theta-\gamma_1\kappa_1\cos\theta]}{[\beta_1\gamma_1\kappa_0\kappa_1-\sin^2\theta]} & \frac{-3\beta_1\kappa_0[3\sin\theta-\gamma_1\kappa_1\cos\theta]}{[\beta_1\gamma_1\kappa_0\kappa_1-\sin^2\theta]} & \frac{3[\beta_1\kappa_0(3\cos\theta+\gamma_1\kappa_1\sin\theta)-\sin\theta]}{[\beta_1\gamma_1\kappa_0\kappa_1-\sin^2\theta]} \end{bmatrix}.$$

The values for $\beta_2, \beta_3, \gamma_2, \gamma_3$ can now be calculated from (A1.1). Substituting the values of δ_i for $i = 0 \ldots 3$ and $\mathbf{A}^{-1}$ the solutions are given by:

$$\beta_2 = \frac{B(\beta_1,\gamma_1)}{D(\beta_1,\gamma_1)}, \quad \beta_3 = \frac{\mathcal{B}(\beta_1,\gamma_1)}{D(\beta_1,\gamma_1)}, \quad \gamma_2 = \frac{G(\beta_1,\gamma_1)}{D(\beta_1,\gamma_1)}, \quad \gamma_3 = \frac{\mathcal{G}(\beta_1,\gamma_1)}{D(\beta_1,\gamma_1)},$$

where,

$$\begin{aligned} B(\beta_1,\gamma_1) =& \frac{1}{3}\big[-60x + 24\beta_1 - 24\cos(\theta)^2\beta_1(1+r) + 60\cos(\theta)^2 x(1+r) \\ &+ 3\cos(\theta)\kappa_1^2\gamma_1^3 - \beta_1^3\kappa_1^2\gamma_1 + 60y\kappa_1\gamma_1(1+r) \\ &+ 60\sin(\theta)\cos(\theta)y(1+r) - 3\kappa_0\sin(\theta)\cos(\theta)\beta_1^2(1+r) - 9\kappa_0\beta_1^2\kappa_1\gamma_1 \\ &- 2\beta_1^3\kappa_1^2\gamma_1 r - \beta_1^3\kappa_1^2\gamma_1 r^2 + \beta_1^3\kappa_1\gamma_1\kappa_0 + 24\beta_1 r \\ &- 60xr - 9\kappa_0\beta_1^2\kappa_1\gamma_1 r + 2\beta_1^3\kappa_1\gamma_1\kappa_0 r + \beta_1^3\kappa_1\gamma_1\kappa_0 r^2 \\ &+ 3\gamma_1^3\cos(\theta)\kappa_1^2 r - 15\sin(\theta)\kappa_1\gamma_1^2(1+r) - \sin(\theta)\gamma_1^3(\kappa_1-\kappa_0) \\ &+ 9\cos^2(\theta)\sin(\theta)\kappa_1\gamma_1(1-\gamma_1)(1+r)\big]/(1+r). \end{aligned}$$

$$
\begin{aligned}
\mathcal{B}(\beta_1,\gamma_1) =& \big[48\gamma_1^2\sin(\theta)\kappa_1 - 9\gamma_1^3\cos(\theta)\kappa_1^2 - 36\beta_1 - \beta_1^3\kappa_0^2 \\
&- 36\beta_1 r + 120xr - 2\cos^2(\theta)\kappa_0^2 r\beta_1^3 + \sin(\theta)\cos(\theta)\kappa_0\beta_1^3 \\
&+ 60\cos^2(\theta)\kappa_0 y\beta_1 + 24\beta_1^2\kappa_0\sin(\theta)\cos(\theta) - \sin(\theta)\cos(\theta)\kappa_1\beta_1^3 \\
&+ 27\sin(\theta)\cos^2(\theta)\kappa_1\gamma_1^2 - 27\cos^2(\theta)\kappa_1\gamma_1\sin(\theta) - 120\cos(\theta)ry\sin(\theta) \\
&+ 48\gamma_1^2\sin(\theta)\kappa_1 r - 9\gamma_1^3\cos(\theta)\kappa_1^2 r + 120x - 3\gamma_1^3\sin(\theta)\kappa_0 + 3\gamma_1^3\sin(\theta)\kappa_1 \\
&- r\beta_1^3\kappa_0^2 + \kappa_0^3\kappa_1 r\beta_1^4\gamma 1 - 15\cos(\theta)\kappa_0\kappa_1 r\beta_1\gamma_1^2 - 15\beta_1\kappa_0\gamma_1^2\kappa_1\cos(\theta) \\
&- \gamma_1^3\kappa_1\beta_1\kappa_0\cos(\theta) + \kappa_0^3\kappa_1\beta_1^4\gamma_1 + \gamma_1^3\kappa_0^2\beta_1\cos(\theta) - 3\sin(\theta)\kappa_0\kappa_1^2\beta_1\gamma_1^3 \\
&+ 60\kappa_0\kappa_1 x\beta_1\gamma_1 - 120\cos^2(\theta)x + 36\cos^2(\theta)\beta_1 - 120\cos^2(\theta)rx \\
&+ 36\cos^2(\theta)r\beta_1 + 60\kappa_0\kappa_1 x\beta_1\gamma_1 - 3\sin(\theta)\kappa_0\kappa_1^2 r\beta_1\gamma_1^3 - 9\kappa_0\kappa_1\beta_1^2\gamma_1 \\
&- 3\kappa_0\kappa_1\beta_1^3\gamma_1 - 180\kappa_1 ry\gamma_1 + 3\kappa_1^2 r^2\beta_1^3\gamma_1 + 6\kappa_1^2 r\beta_1^3\gamma_1 - 9\kappa_0\kappa_1 r\beta_1^2\gamma_1 \\
&- 3\kappa_0\kappa_1 r^2\beta_1^3\gamma_1 - 6\kappa_0\kappa_1 r\beta_1^3\gamma_1 + 24\cos(\theta)r\beta_1^2\kappa_0\sin(\theta) \\
&+ 9\cos^3(\theta)\kappa_0\kappa_1\beta_1\gamma_1 + 60\cos^2(\theta)\kappa_0 ry\beta_1 + 2\sin(\theta)\cos(\theta)\kappa_0 r\beta_1^3 \\
&- 60\sin(\theta)\cos(\theta)\kappa_0 x\beta_1 + \sin(\theta)\cos(\theta)\kappa_0 r^2\beta_1^3 - 9\cos^3(\theta)\kappa_0\kappa_1\beta_1\gamma_1^2 \\
&- 27\cos^2(\theta)\kappa_1 r\gamma_1\sin(\theta) + 27\sin(\theta)\cos^2(\theta)\kappa_1 r\gamma_1^2 - 2\sin(\theta)\cos(\theta)\kappa_1 r\beta_1^3 \\
&- \sin(\theta)\cos(\theta)\kappa_1 r^2\beta_1^3 - 3\kappa_1^2\beta_1^3\gamma_1 - 180\kappa_1 y\gamma_1 - 120\cos(\theta)y\sin(\theta) \\
&- 2\cos^2(\theta)\kappa_0^2\beta_1^3 - 60\sin(\theta)\cos(\theta)\kappa_0 rx\beta_1 + 9\cos^3(\theta)\kappa_0\kappa_1 r\beta_1\gamma_1 \\
&- 9\cos^3(\theta)\kappa_0\kappa_1 r\beta_1\gamma_1^2\big]/(1+r).
\end{aligned}
$$

$$
\begin{aligned}
G(\beta_1,\gamma_1) =& \frac{1}{3}\big[- 24\gamma_1(1+r) + 60\sin(\theta)y(1+r) - 3\cos(\theta)\kappa_0^2\beta_1^3(1+r) \\
&+ 15\sin(\theta)\kappa_0\beta_1^2(1+r) + 60\cos(\theta)y\beta_1\kappa_0(1+r) - 60x\sin(\theta)\beta_1\kappa_0(1+r) \\
&+ 3\sin(\theta)\cos(\theta)\kappa_1\gamma_1^2(1+r) + \sin(\theta)\beta_1^3\kappa_0 - \sin(\theta)\beta_1^3\kappa_1
\end{aligned}
$$

$$
\begin{aligned}
&+ \gamma_1^3\beta_1\kappa_0^2 - 2\sin(\theta)\beta_1^3\kappa_1 r - \sin(\theta)\beta_1^3\kappa_1 r^2 \\
&+ 2\sin(\theta)\beta_1^3\kappa_0 r + \sin(\theta)\beta_1^3\kappa_0 r^2 - \gamma_1^3\beta_1\kappa_0\kappa_1 \\
&+ 9\kappa_1\gamma_1^2\beta_1\kappa_0(1+r) + 24\cos(\theta)^2\gamma_1(1+r) \\
&+ 9\cos^2(\theta)\kappa_0\kappa_1\beta_1\gamma_1(1-\gamma_1)(1+r)\big]/(1+r).
\end{aligned}
$$

$$
\begin{aligned}
\mathcal{G}(\beta_1,\gamma_1) = &-\big[-3\kappa_0\beta_1^3\sin(\theta) + 3\kappa_1\beta_1^3\sin(\theta) - 120yr\sin(\theta) \\
&+ 9\beta_1^3\kappa_0^2\cos(\theta) + 36\gamma_1 + 36\gamma_1 r - 120y\sin(\theta) - 27\cos^2(\theta)\kappa_0\kappa_1\beta_1\gamma_1 \\
&+ 9\gamma_1^2\kappa_1 r\beta_1\kappa_0 - 180\cos(\theta)r\beta_1\kappa_0 y + 180\beta_1\kappa_0 x\sin(\theta)r \\
&- 33\gamma_1^2\cos(\theta)\kappa_1\sin(\theta)r + 9\beta_1\kappa_0\gamma_1^2\kappa_1 + 3\gamma_1^3\kappa_1\beta_1\kappa_0 - 6\kappa_0 r\beta_1^3\sin(\theta) \\
&+ 6\kappa_1 r\beta_1^3\sin(\theta) - 48\beta_1^2\kappa_0 r\sin(\theta) + 180\beta_1\kappa_0 x\sin(\theta) \\
&- 33\gamma_1^2\cos(\theta)\kappa_1\sin(\theta) + 9\cos(\theta)r\beta_1^3\kappa_0^2 - 3\kappa_0 r^2\beta_1^3\sin(\theta) \\
&+ 3\kappa_1 r^2\beta_1^3\sin(\theta) - 180\beta_1\kappa_0\cos(\theta)y + 2\gamma_1^3\cos^2(\theta)\kappa_1^2 r - \kappa_1^2\beta_1^3\gamma_1\cos(\theta) \\
&+ 60y\gamma_1\cos(\theta)\kappa_1 + \sin(\theta)\cos(\theta)\kappa_0\gamma_1^3 - \sin(\theta)\cos(\theta)\kappa_1\gamma_1^3 \\
&+ 9\sin(\theta)\cos(\theta)\kappa_1\gamma_1 - \kappa_0\kappa_1^3\beta_1\gamma_1^4 - 48\beta_1^2\kappa_0\sin(\theta) - 3\gamma_1^3\kappa_0^2\beta_1 \\
&- 36\cos^2(\theta)\gamma_1 - 36\cos^2(\theta)r\gamma_1 - \kappa_0\kappa_1^3 r\beta_1\gamma_1^4 + 3\sin(\theta)\kappa_0^2\kappa_1\beta_1^3\gamma_1 \\
&+ 9\sin(\theta)\cos(\theta)\kappa_1 r\gamma_1 - 2\kappa_1^2 r\beta_1^3\gamma_1\cos(\theta) - \kappa_1^2 r^2\beta_1^3\gamma_1\cos(\theta) \\
&+ 15\beta_1^2\kappa_0\gamma_1\cos(\theta)\kappa_1 + \kappa_0\beta_1^3\gamma_1\cos(\theta)\kappa_1 + 60yr\gamma_1\cos(\theta)\kappa_1 \\
&+ 27\cos^2(\theta)\kappa_0\kappa_1\beta_1\gamma_1^2 + \kappa_1^2\gamma_1^3 + \kappa_1^2 r\gamma_1^3 + 2\gamma_1^3\cos^2(\theta)\kappa_1^2 \\
&+ 9\cos(\theta)r\beta_1\kappa_0\gamma_1^3\sin(\theta)\kappa_1^2 - 60\beta_1\kappa_0 x r\gamma_1\cos(\theta)\kappa_1 \\
&- 9\sin(\theta)\cos(\theta)\kappa_0\kappa_1^2 r\beta_1\gamma_1^2 - 60\sin(\theta)\kappa_0\kappa_1 ry\beta_1\gamma_1 \\
&+ 9\gamma_1^3\cos(\theta)\kappa_1^2\beta_1\kappa_0\sin(\theta) + 15\beta_1^2\kappa_0 r\gamma_1\cos(\theta)\kappa_1
\end{aligned}
$$

$$
\begin{aligned}
&- 60\beta_1\kappa_0 x\gamma_1\cos(\theta)\kappa_1 + \kappa_0 r^2\beta_1^3\gamma_1\cos(\theta)\kappa_1 + 27\cos^2(\theta)\kappa_0\kappa_1 r\beta_1\gamma_1^2 \\
&+ 2\kappa_0 r\beta_1^3\gamma_1\cos(\theta)\kappa_1 + 3\sin(\theta)\kappa_0^2\kappa_1 r\beta_1^3\gamma_1 - 9\sin(\theta)\cos(\theta)\kappa_0\kappa_1^2\beta_1\gamma_1^2 \\
&- 60\sin(\theta)\kappa_0\kappa_1 y\beta_1\gamma_1 - 27\cos^2(\theta)\kappa_0\kappa_1 r\beta_1\gamma_1\big]/(1+r).
\end{aligned}
$$

$$
D(\beta_1,\gamma_1) = \beta_1\gamma_1\kappa_0\kappa_1 - \sin^2(\theta).
$$

Appendix B

A convergent subdivision routine

This appendix presents a subdivision scheme which is only one of many possible schemes. Its presence in this thesis is used purely to show that such a scheme does exist.

Given a GCS that lies outside the bounds stated in section 6.3.1, the following subdivision scheme yields a set of GCS curves that lie within the bounds. The original GCS can be formed by joining these subdivided curves as outline in section 6.3.1.

From section 6.3.2: $\bar{\theta}_1, \bar{\theta}_2 < \bar{\theta}$, $|r_1|, |r_2| < |r|$ and $|t_1|, |t_2| < |t|$, thus subdivision decreases the values of $\bar{\theta}, |r|$ and $|t|$. It is therefore enough to consider each parameter in turn, developing a subdivision process that yields $\bar{\theta} \leq \frac{\pi}{2}$, $\frac{-8}{9} \leq r \leq 8$ and then $|t| \leq \pi$.

Case 1: When $\theta > \frac{\pi}{2}$

Let $\bar{\theta} = n\frac{\pi}{2}$ and $N = \lceil n \rceil$ where "$\lceil n \rceil$" is the smallest integer not less than n. Then it is possible to subdivide the curve into N segments each with uniform total winding angle $\bar{\theta}_i = \frac{\bar{\theta}}{N} \leq \frac{\pi}{2}$ for $i = 1 \ldots N$. This is achieved by solving

$$\int_{\lambda_{i-1}}^{\lambda_i} |\kappa(s)| ds = \frac{\bar{\theta}}{N} \qquad \text{for } i = 1 \ldots N,$$

where λ_i are the subdivision parameters and $\lambda_0 = 0$ and $\lambda_N = 1$. Note that each λ_i can always be found using a suitable numerical procedure since the cumulative integral of a positive function behaves monotonically.

Thus each segment will satisfy $\bar{\theta} \leq \frac{\pi}{2}$ and thus $|\theta| \leq \frac{\pi}{2}$.

Case 2: When $|u - \frac{1}{2}| > 0.4$

When $|u - \frac{1}{2}| > 0.4$ either $r < \frac{-8}{9}$ or $r > 8$. Each case is examined in turn.

As an outline, the subdivision process works by decreasing the size of the shape factor $|r|$ for one of the subdivided curves, as to lie on the bound, whilst simultaneously decreasing the size of the other curve's shape factor. After an appropriate number of subdivisions all GCS curves lie within the bounds.

For the first case, assume $r < \frac{-8}{9}$ and let $r = -1 + \delta$ then $0 < \delta < \frac{1}{9}$. Subdividing at $\lambda = \frac{-8}{9r}$:

$$r_1 = \frac{-8}{9}$$
$$r_2 = 9r + 8 = -1 + 9\delta$$

If r_2 is not within the bounds subdivision is repeated until $r_2 > \frac{-8}{9}$ as detailed below.

Labelling the sequential values of δ after subdivision as δ_i, then clearly $\delta_{i+1} = 9\delta_i$. Thus after I subdivisions $\delta_I = 9^I\delta$.

Let $n = -(1 + \log_9(\delta))$ then $\frac{1}{9} = 9^n\delta$. Let $N = \lceil n \rceil$ then after N subdivisions $\delta_i < \frac{1}{9}$ for $i = 0 \ldots N - 1$ and $\frac{1}{9} \leq \delta_N < 1$. Thus $\frac{-8}{9} \leq r_N < 0$.

By construction $r_i = \frac{-8}{9}$ for $i = 1 \ldots N - 1$ and $\frac{-8}{9} \leq r_N < 0$ thus each segment satisfies $|u - \frac{1}{2}| \leq 0.4$. Furthermore N subdivisions are required, where $N = \lceil -(1 + \log_9(1 + r)) \rceil$.

Now assume $r > 8$ and let $r = -1 + \frac{1}{\delta}$ then $0 < \delta < \frac{1}{9}$. Choosing $\lambda = \frac{r-8}{9r}$:

$$r_1 = \frac{r-8}{9} = \frac{-1 + \frac{1}{\delta} - 8}{9} = \frac{1 - 9\delta}{9\delta} = -1 + \frac{1}{9\delta}$$
$$r_2 = 8$$

Again this process is repeated until $r_1 < 8$ as detailed below.

Similarly to the first case, labeling sequential values of δ as δ_i, then $\delta_{i+1} = 9\delta_i$ and thus $\delta_I = 9^I\delta$.

Let $n = -(1 + \log_9(\delta))$ then $\frac{1}{9} = 9^n\delta$. Let $N = \lceil n \rceil$ then after N subdivisions $\delta_i < \frac{1}{9}$ for $i = 0 \ldots N-1$ and $\frac{1}{9} \leq \delta_N < 1$. Thus $0 < r_N \leq 8$.

By construction $r_i = 8$ for $i = 1 \ldots N-1$ and $0 < r_N \leq 8$ thus each segment satisfies $|u - \frac{1}{2}| \leq 0.4$. Furthermore N subdivisions are required, where $N = \lceil (-1 + \log_9(1 + r)) \rceil$.

Case 3: When $|t| > \pi$

Let $|t| = n\pi$ and $N = \lceil n \rceil$. The curve can be split into N segments each of equal length. Then the t_i for each subdivision is

$$t_i = \left(\kappa\left(\frac{i-1}{N}\right) - \kappa\left(\frac{i}{N}\right)\right)\frac{1}{N} \qquad \text{for } i = 1 \ldots N,$$

and thus

$$|t_i| = \left|\frac{\left(\kappa\left(\frac{i-1}{N}\right) - \kappa\left(\frac{i}{N}\right)\right)}{N}\right| \leq \left|\frac{(\kappa_0 - \kappa_1)}{N}\right| = \left|\frac{t}{N}\right| \leq \pi.$$

Therefore each segment satisfies $|t| \leq \pi$. This completes the subdivision routine.

Appendix C

Isometries of the curvature

This appendix looks at mathematical conditions on the curvature profile of a curve which yield isometric curves. Two variations of a curvature profile are studied in turn, $\kappa(s) \leftrightarrow -\kappa(s)$ and $\kappa(s) \leftrightarrow -\kappa(A - s)$.

Curve Reflections ($\kappa(s) \leftrightarrow -\kappa(s)$)

It will be shown that given a curve (I) with curvature profile $\kappa(s)$ for $s \in [0, A]$ and another curve (II) with a corresponding curvature profile $\tilde{\kappa}(s) = -\kappa(s)$ then the two curves are isometric. Since the curves need only be identical up to a translation, rotation and reflection the initial curves position and direction can be arbitrary. For simplicity let the origin be $(0, 0)$ and the initial tangent vector point along the x-axis.

It will be shown that the two curves are identical up to a reflection in the x-axis. This validates the claim that the two curves are isometric (see Fig C2.1).

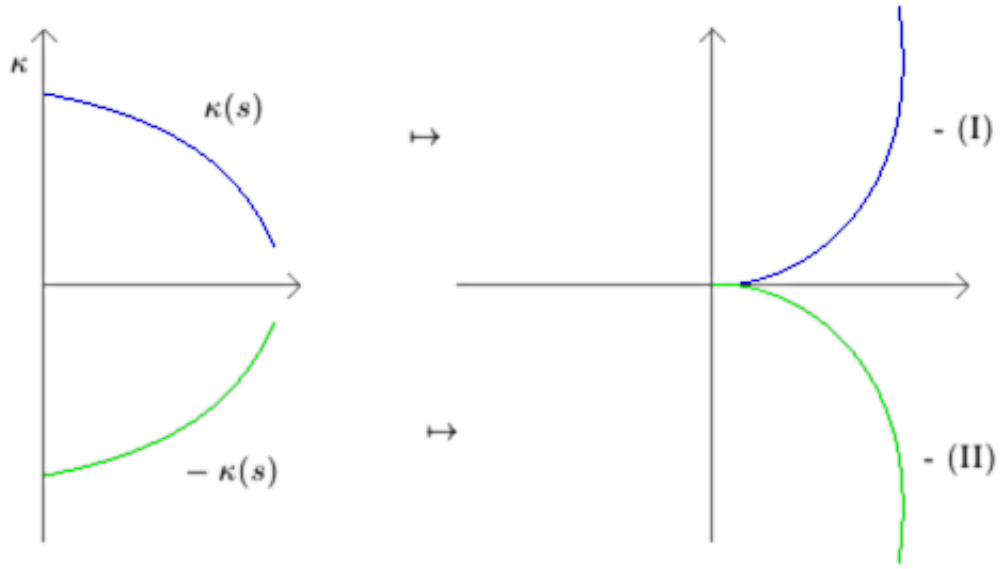

Figure C.1: The two curves are symmetrical.

Proof of isometry

The parametric representation for (I) is

$$\begin{bmatrix} x(t) \\ y(t) \end{bmatrix} = \begin{bmatrix} \int_0^t \cos(\int_0^\sigma \kappa(\tau)d\tau)d\sigma \\ \int_0^t \sin(\int_0^\sigma \kappa(\tau)d\tau)d\sigma \end{bmatrix} \quad \text{for } t \in [0, A].$$

For (II) the parameterisation is:

$$\begin{bmatrix} \tilde{x}(t) \\ \tilde{y}(t) \end{bmatrix} = \begin{bmatrix} \int_0^t \cos(\int_0^\sigma -\kappa(\tau)d\tau)d\sigma \\ \int_0^t \sin(\int_0^\sigma -\kappa(\tau)d\tau)d\sigma \end{bmatrix} = \begin{bmatrix} \int_0^t \cos(-\int_0^\sigma \kappa(\tau)d\tau)d\sigma \\ \int_0^t \sin(-\int_0^\sigma \kappa(\tau)d\tau)d\sigma \end{bmatrix} =$$

$$\begin{bmatrix} \int_0^t \cos(\int_0^\sigma \kappa(\tau)d\tau)d\sigma \\ -\int_0^t \sin(\int_0^\sigma \kappa(\tau)d\tau)d\sigma \end{bmatrix} \text{ for } t \in [0, A].$$

Therefore, $x(t) = \tilde{x}(t)$ and $y(t) = -\tilde{y}(t)$. Since the parameter is arc-length for both curves it is enough to say the curves are symmetrical about the x-axis and thus isometric.

Parametric Reflection ($\kappa(s) \leftrightarrow -\kappa(A - s)$)

It will be shown that given a curve (I) with curvature profile $\kappa(s)$ for $s \in [0, A]$ and another curve (II) with a corresponding curvature profile $\tilde{\kappa}(s) = -\kappa(A - s)$ then the two curves are isometric. Since the curves need only be identical up to a translation, rotation and reflection the initial curves position and direction can be arbitrary. For simplicity let the origin be $(0, 0)$ and the initial tangent vector point along the x-axis.

An outline of the proof can summarised by the following. Both curves are plotted and a rotation, $\pi + \psi$, followed by a translation, (x, y), is applied to one of the curves and they are then shown to plot the same curve. (It follows that ψ corresponds to the total winding angle of the curve and (x, y) corresponds to the end points. See figures C.2 and C.3.)

Proof of isometry

Parameterising the first curve (I) gives:

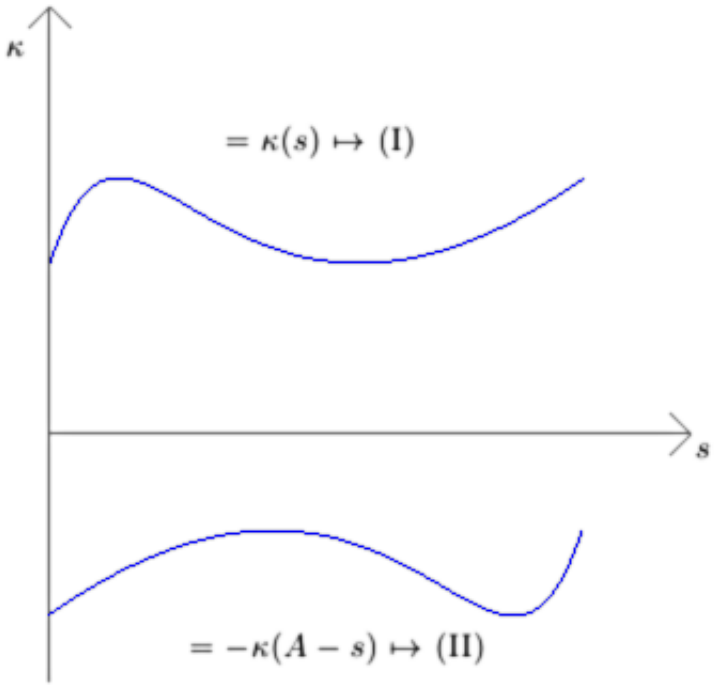

Figure C.2: $\kappa(s)$ vs. $-\kappa(A-s)$.

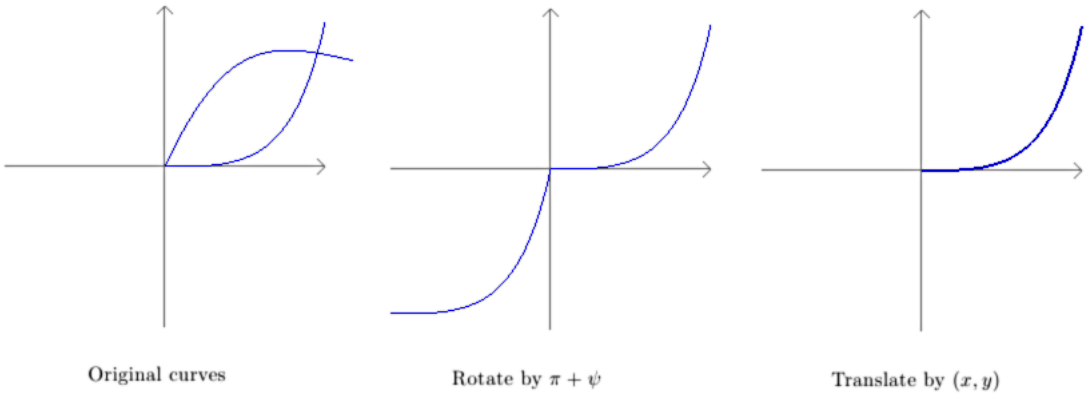

Figure C.3: The two curves are geometrically equivalent.

$$\begin{bmatrix} x(t) \\ y(t) \end{bmatrix} = \begin{bmatrix} \int_0^t \cos(\int_0^\sigma \kappa(\tau)d\tau)d\sigma \\ \int_0^t \sin(\int_0^\sigma \kappa(\tau)d\tau)d\sigma \end{bmatrix} \quad \text{for } t \in [0, A].$$

For (II) the parameterisation is:

$$\begin{bmatrix} \tilde{x}(t) \\ \tilde{y}(t) \end{bmatrix} = \begin{bmatrix} \int_0^t \cos(\int_0^\sigma -\kappa(A-\tau)d\tau)d\sigma \\ \int_0^t \sin(\int_0^\sigma -\kappa(A-\tau)d\tau)d\sigma \end{bmatrix} \quad \text{for } t \in [0, A].$$

Applying the rotation gives:

$$\begin{bmatrix} \tilde{x}(t) \\ \tilde{y}(t) \end{bmatrix} = \begin{bmatrix} \int_0^t \cos(\pi + \psi + \int_0^\sigma -\kappa(A-\tau)d\tau)d\sigma \\ \int_0^t \sin(\pi + \psi + \int_0^\sigma -\kappa(A-\tau)d\tau)d\sigma \end{bmatrix}.$$

Finally, applying the translation gives:

$$\begin{bmatrix} \tilde{x}(t) \\ \tilde{y}(t) \end{bmatrix} = \begin{bmatrix} \int_0^t \cos(\pi + \psi + \int_0^\sigma -\kappa(A-\tau)d\tau)d\sigma + \int_0^A \cos(\int_0^\sigma \kappa(\tau)d\tau)d\sigma \\ \int_0^t \sin(\pi + \psi + \int_0^\sigma -\kappa(A-\tau)d\tau)d\sigma + \int_0^A \sin(\int_0^\sigma \kappa(\tau)d\tau)d\sigma \end{bmatrix}.$$

Consider $\int_0^\sigma -\kappa(A-\tau)d\tau$ then by integration by substitution and letting $f(\tau) = \kappa(\tau)$ and $g(\tau) = A - \tau$

$$\int_0^\sigma -\kappa(A-\tau)d\tau = \int_a^b f(g(\tau))g'(\tau)d\tau = \int_{g(a)}^{g(b)} f(\tau)d\tau = \int_A^{A-\sigma} \kappa(\tau)d\tau.$$

Hence,

$$\int_0^\sigma -\kappa(A-\tau)d\tau = \int_A^{A-\sigma} \kappa(\tau)d\tau.$$

Now, $\psi = \int_0^A \kappa(\tau)d\tau$

so $\psi + \int_0^\sigma -\kappa(A-\tau)d\tau = \int_0^A \kappa(\tau)d\tau + \int_A^{A-\sigma} \kappa(\tau)d\tau = \int_0^{A-\sigma} \kappa(\tau)d\tau.$

Also since $\cos(\alpha+\pi) = -\cos(\alpha)$ and $\sin(\alpha+\pi) = -\sin(\alpha)$ then:

$$\begin{bmatrix} \tilde{x}(t) \\ \tilde{y}(t) \end{bmatrix} = \begin{bmatrix} -\int_0^t \cos(\int_0^{A-\sigma} \kappa(\tau)d\tau)d\sigma + \int_0^A \cos(\int_0^\sigma \kappa(\tau)d\tau)d\sigma \\ -\int_0^t \sin(\int_0^{A-\sigma} \kappa(\tau)d\tau)d\sigma + \int_0^A \sin(\int_0^\sigma \kappa(\tau)d\tau)d\sigma \end{bmatrix}.$$

Consider the integral

$$-\int_0^t \cos(\int_0^{A-\sigma} \kappa(\tau)d\tau)d\sigma.$$

Integrating by substitution, with

$$f(t) = \cos(\int_0^t \kappa(\tau)d\tau), \quad g(t) = A - t \quad \Rightarrow$$

$$\begin{aligned} -\int_0^t \cos(\int_0^{A-\sigma} \kappa(\tau)d\tau)d\sigma &= \int_a^b f(g(t))g'(t)dt \\ &= \int_{g(a)}^{g(b)} f(t)dt = \int_A^{A-t} \cos(\int_0^\sigma \kappa(\tau)d\tau)d\sigma. \end{aligned}$$

Hence,

$$-\int_0^t \cos(\int_0^{A-\sigma} \kappa(\tau)d\tau)d\sigma = \int_A^{A-t} \cos(\int_0^\sigma \kappa(\tau)d\tau)d\sigma.$$

Similarly,

$$-\int_0^t \sin(\int_0^{A-\sigma} \kappa(\tau)d\tau)d\sigma = \int_A^{A-t} \sin(\int_0^\sigma \kappa(\tau)d\tau)d\sigma.$$

Therefore,

$$\begin{bmatrix} \tilde{x}(t) \\ \tilde{y}(t) \end{bmatrix} = \begin{bmatrix} \int_A^{A-t} \cos(\int_0^\sigma \kappa(\tau)d\tau)d\sigma + \int_0^A \cos(\int_0^\sigma \kappa(\tau)d\tau)d\sigma \\ \int_A^{A-t} \sin(\int_0^\sigma \kappa(\tau)d\tau)d\sigma + \int_0^A \sin(\int_0^\sigma \kappa(\tau)d\tau)d\sigma \end{bmatrix}.$$

Hence,

$$\begin{bmatrix} \tilde{x}(t) \\ \tilde{y}(t) \end{bmatrix} = \begin{bmatrix} \int_0^{A-t} \cos(\int_0^\sigma \kappa(\tau)d\tau)d\sigma \\ \int_0^{A-t} \sin(\int_0^\sigma \kappa(\tau)d\tau)d\sigma \end{bmatrix}.$$

Comparing, $\begin{bmatrix} \tilde{x}(t) \\ \tilde{y}(t) \end{bmatrix} = \begin{bmatrix} x(A-t) \\ y(A-t) \end{bmatrix}$ thus they define the same curve, and hence (I) and (II) are isometric.

Note that this proves that it is possible to draw the curve backwards. In essence the curve starts at the end point and is traversed backwards through the curvature plot. This provides an interesting geometric interpretation of the invariance between the two plots.